



**GABRIELA DUARTE  
SOBRAL de JESUS**

**APROXIMAÇÃO NUMÉRICA DE VALORES  
PRÓPRIOS DE MATRIZES REAIS**





**GABRIELA DUARTE  
SOBRAL de JESUS**

**APROXIMAÇÃO NUMÉRICA DE VALORES  
PRÓPRIOS DE MATRIZES REAIS**

Dissertação apresentada à Universidade de Aveiro para cumprimento dos requisitos necessários à obtenção do grau de Mestre em Matemática e Aplicações, área de especialização Matemática Empresarial e Tecnológica, realizada sob a orientação científica da Doutora Isabel Alexandra Vieira Brás, Professora Auxiliar do Departamento de Matemática da Universidade de Aveiro e do Doutor Jorge Manuel Sá Esteves, Professor Auxiliar do Departamento de Matemática da Universidade de Aveiro.



**o júri / the jury**

presidente / president

**Professora Doutora Isabel Maria Simões Pereira**

Professora Auxiliar do Departamento de Matemática da Universidade de Aveiro

vogais / examiners committee

**Professora Doutora Ana Catarina Santos Carapito**

Professora Auxiliar da Universidade da Beira Interior

**Professor Doutor Jorge Manuel Sá Esteves**

Professor Auxiliar do Departamento de Matemática da Universidade de Aveiro (orientador)

**Professora Doutora Isabel Alexandra Vieira Brás**

Professora Auxiliar do Departamento de Matemática da Universidade de Aveiro (co-orientadora)



## **agradecimentos / acknowledgements**

Não é fácil colocar por palavras aquilo que se sente no final de mais um percurso...

Começo por agradecer aos meus orientadores, Professora Dra. Isabel Brás e Professor Dr. Sá Esteves, pelo incansável apoio, compreensão e paciência, por todas as horas disponibilizadas para que este trabalho chegasse a bom porto.

Agradeço, também, à minha colega de mestrado, Sara Sá, pela divertida companhia e apoio neste trajeto, e à minha amiga, Catarina Marques, que sempre acreditou em mim.

Obrigada à minha família por estar sempre do meu lado.

Um agradecimento do fundo do coração às minhas filhotas, Eliana e Leonor, que, apesar da idade, compreenderam a importância das horas que a mãe não passou a brincar com elas. Prometo compensar!

Finalmente, muito obrigada ao meu Tó por compreender o significado desta etapa na minha vida e estar incondicionalmente ao meu lado.





**Palavras-chave**

Matrizes Reais, Valores Próprios, Zeros de Polinómios, Métodos Numéricos.

**Resumo**

O problema numérico de calcular os valores próprios de matrizes com entradas reais é introduzido referindo algumas notas históricas e as suas múltiplas aplicações. Noções básicas de teoria de matrizes são apresentadas. Alguns métodos clássicos e básicos de localização de valores próprios são apresentados, usando normas matriciais e recorrendo aos círculos de Gershgorin. O problema de calcular o polinómio característico de uma matriz é abordado através dos métodos de Danilevsky, de Krylov, de Leverrier e também do método dos coeficientes indeterminados. Uma vez que um valor próprio de uma matriz é uma raiz do seu polinómio característico, descrevem-se os métodos numéricos de Newton-Horner, de Müller e de Bairstow, para o cálculo de raízes de polinómios. Por último, são estudados métodos iterativos matriciais, que são os que atualmente se preferem, para o cálculo dos valores próprios de uma matriz. Nesse estudo, contemplam-se o Método da Potência (diretas e inversas), o Método QR (nas variantes básica e Hessenberg-QR) e o Método de Jacobi. Tecem-se as principais conclusões da dissertação e termina-se com um conjunto de pistas para eventual trabalho adicional. Anexam-se alguns métodos históricos de localização de zeros de polinómios e, também, o código das rotinas MATLAB®, usadas para as simulações numéricas dos exemplos que atravessam todo o texto.



**Keywords**

Real Matrices, Eigenvalues, Polynomial Roots, Numerical Methods.

**Abstract**

The numerical problem of computing the eigenvalues of a real matrix is introduced with reference to some historical notes and its multiple applications. Basic notions of matrix theory are presented. Some classic and basic methods of eigenvalues localization are presented, using matrix norms and Gershgorin circles. The problem of computing the characteristic polynomial of a matrix is addressed through the methods of Danilevsky, Krylov, Leverrier and, also the undetermined coefficient method. Once that a matrix eigenvalue is a root of its characteristic polynomial, we describe the numerical methods of Newton-Horner, Muller and Bairstow, to calculate polynomial roots. Finally, we study matrix iterative methods that, nowadays, are preferred for computing matrix eigenvalues. In this thesis, we include the Power Method (Direct and Inverse), the QR Method (basic and Hessenberg-QR variants) and the Jacobi Method. We establish the main conclusions of this thesis and end with a set of clues for eventual additional work. We append some historical methods of polynomial roots localization and, also, the codes of MATLAB<sup>®</sup> routines used for the numerical simulations in the examples given throughout the text



“De que me irei ocupar no céu, durante toda a Eternidade, se não me derem uma infinidade de problemas de Matemática para resolver?”

– *Augustin Louis Cauchy*



# Conteúdo

<b>Conteúdo</b>	<b>i</b>
<b>Tabela de Símbolos</b>	<b>1</b>
<b>1 Introdução</b>	<b>3</b>
1.1 Enquadramento . . . . .	3
1.2 Notas Históricas . . . . .	5
1.3 Aplicações Práticas . . . . .	7
1.4 Objetivos Gerais do Trabalho . . . . .	8
1.5 Organização da Dissertação . . . . .	9
<b>2 Preliminares</b>	<b>11</b>
2.1 Normas matriciais . . . . .	11
2.2 Valores e Vetores Próprios de uma matriz . . . . .	14
2.3 Transformações de Semelhança . . . . .	16
2.4 Matrizes de Householder . . . . .	19
2.5 Matrizes de Givens . . . . .	20
<b>3 Localização de valores próprios</b>	<b>23</b>
3.1 Com recurso a normas . . . . .	23
3.2 Os Círculos de Gershgorin . . . . .	29
3.3 Exemplos . . . . .	32
<b>4 Obtenção do polinómio caraterístico, dada uma matriz</b>	<b>37</b>
4.1 O método de Danilevsky . . . . .	37
4.2 O método de Krylov . . . . .	44
4.3 O método de Leverrier . . . . .	46
4.4 O método dos coeficientes indeterminados . . . . .	47
<b>5 Métodos Numéricos de Aproximação de Zeros de Polinómios</b>	<b>51</b>
5.1 Método de Newton-Horner . . . . .	52
5.2 Método de Müller . . . . .	57
5.3 Método de Bairstow . . . . .	60

<b>6</b>	<b>Métodos Matriciais Iterativos de Aproximação de Valores Próprios</b>	<b>67</b>
6.1	O Método da Potência . . . . .	67
6.1.1	Acerca do Método das Potências Diretas . . . . .	75
6.1.2	O Método das Potências Inversas . . . . .	77
6.1.3	Acerca do Método das Potências Inversas . . . . .	78
6.2	O Método QR . . . . .	79
6.2.1	Fatorização QR de matrizes quadradas . . . . .	80
6.2.2	O Método QR Básico . . . . .	89
6.2.3	O Método Hessenberg-QR . . . . .	91
6.2.4	A convergência do Método QR . . . . .	96
6.3	O Método de Jacobi . . . . .	101
6.3.1	Convergência do Método de Jacobi . . . . .	105
<b>7</b>	<b>Conclusões</b>	<b>109</b>
7.1	Síntese do Trabalho Apresentado . . . . .	109
7.2	Pistas para Trabalho Adicional . . . . .	110
<b>A</b>	<b>Localização de Zeros de Polinómios</b>	<b>113</b>
A.1	Número de raízes reais de um polinómio . . . . .	113
A.1.1	Regra de Budan-Fourier . . . . .	114
A.1.2	Regra dos Sinais de Descartes . . . . .	114
A.1.3	Exemplo . . . . .	114
A.2	Localização de raízes de um polinómio . . . . .	115
<b>B</b>	<b>Rotinas MATLAB®</b>	<b>119</b>
B.1	Métodos Numéricos de Aproximação de Zeros de Polinómios . . . . .	119
B.2	Métodos Matriciais Iterativos de Aproximação de Valores Próprios . . . . .	121
	<b>Bibliografia</b>	<b>125</b>



# Tabela de Símbolos

$\mathbb{C}^n$	Espaço linear sobre $\mathbb{C}$ dos vectores de dimensão $n$ , com elementos em $\mathbb{C}$ , escritos sob a forma de matrizes-coluna.
$\mathbb{C}^{n \times n}$	Espaço vectorial sobre $\mathbb{C}$ das matrizes, $n \times n$ , com entradas em $\mathbb{C}$ .
$\ \cdot\ $	Norma vetorial em $\mathbb{C}$ .
$   \cdot   $	Norma matricial em $\mathbb{C}^{n \times n}$ ou norma matricial induzida por uma norma vetorial $\ \cdot\ $ .
$I_n$	Matriz identidade de ordem $n$ .
$A^*$	Matriz transconjugada de $A$ .
$A^T$	Matriz transposta de $A$ .
$A^{-1}$	Matriz inversa de $A$ .
$A^k$	Matriz potência de $A$ , de ordem $k$ .
$p_A(\lambda)$	Polinómio caraterístico de $A$ .
$\sigma(A)$	Espetro de $A$ .
$\rho(A)$	Raio espectral de $A$ .
$\alpha_A(\lambda_i)$	Multiplicidade algébrica do valor próprio $\lambda_i$ , de $A$ .
$S_A(\lambda)$	Subespaço próprio de $A$ .
$\gamma_A(\lambda)$	Multiplicidade geométrica do valor próprio $\lambda$ , de $A$ .
$\text{tr}(A)$	Traço de $A$ .
$\det(A)$	Determinante de $A$ .

$C_A$	Matriz companheira da matriz $A$ .
$\mathcal{P}_{p,q}$	$(p, q)$ -plano em $\mathbb{R}^n$
$G(p, q, \theta)$	Matriz de Givens no $(p, q)$ -plano associada ao ângulo $\theta$ .
$\ell(A)$	Limite inferior de uma matriz $A$ , relativamente a uma norma vetorial.
$G_i(A)$	Círculo de Gershgorin, por linha, no plano complexo, centrado em $a_{ii}$ de raio $r_i(A) = \sum_{j=1, j \neq i}^n  a_{ij} $ , com $A = [a_{ij}]$ matriz em $\mathbb{C}^{n \times n}$ .
$G(A)$	Região de Gershgorin de $A$ , definida por $\cup_{i=1}^n G_i(A)$ .
$G'(A)$	Região de Gershgorin de $A$ , definida por $\cup_{j=1}^n C_j$ , onde $C_j$ é um círculo, do plano complexo, com centro em $ a_{jj} $ e raio $r_j(A) = \sum_{i=1, i \neq j}^n  a_{ij} $ , com $A = [a_{ij}]$ matriz em $\mathbb{C}^{n \times n}$ .

# Capítulo 1

## Introdução

### 1.1 Enquadramento

Um vetor próprio de uma matriz quadrada é um vetor não nulo que, quando pré-multiplicado pela matriz, gera outro vetor que é um seu múltiplo escalar. Esse fator multiplicativo é o valor próprio (ou autovalor) associado a esse vetor próprio. De uma forma mais geral, suponhamos um espaço vetorial  $V$ , de dimensão finita, sobre um corpo  $\mathbb{K}$  e um operador linear  $A : V \rightarrow V$ . Se existir  $x \in V \setminus \{0\}$  tal que  $Ax = \lambda x$ , onde  $\lambda \in \mathbb{K}$ , o vetor  $x$  é chamado vetor próprio e o escalar  $\lambda$  é o correspondente valor próprio de  $A$ . As matrizes que consideraremos neste trabalho, são matrizes com entradas reais que, por questões de comodidade teórica, serão consideradas em  $\mathbb{K} = \mathbb{C}$ . Note-se que, se conhecermos um valor próprio, então é fácil calcular o correspondente vetor próprio, encontrando a solução não nula do sistema linear:

$$Ax = \lambda x \Leftrightarrow (A - \lambda I_n)x = 0, \quad x \neq 0. \quad (1.1)$$

O sistema (1.1) só terá uma solução não nula se a matriz  $(A - \lambda I_n)$  for singular. Assim, um valor próprio de  $A$  é um número  $\lambda$  tal que, se for subtraído de cada entrada na diagonal de  $A$ , converte  $A$  numa matriz singular. Este facto implica que o conjunto de todos os valores próprios de uma dada matriz quadrada  $A$  é constituído pelos números que “*captam*” algumas das propriedades mais importantes daquela matriz. Chama-se a esse conjunto de números o *espectro* da matriz  $A$ .

As transformações lineares em espaços vetoriais podem ter aplicação em múltiplas situações. Para dar exemplos práticos, fáceis de perceber, podemos imaginar os seguintes:

- Coloquemos uma moeda na vertical e façamo-la girar. O objeto sofre uma transformação rotacional mas essa rotação faz-se num eixo vertical, que é uma direção importante para estudar o movimento. Essa direção será o vetor próprio da transformação. Na verdade, os pontos da moeda que estão nessa direção não sofrem nenhuma mudança de posição. Ou seja, o valor próprio associado é um.
- Tomemos uma banda elástica e estiquemo-la na direção longitudinal, duplicando o seu comprimento. O objeto sofre uma transformação mas na direção longitudinal

tudo é ampliado, de uma maneira uniforme, para o dobro. Neste caso, a direção longitudinal define o vetor próprio, sendo o valor próprio correspondente igual a dois ([28]).

Estes exemplos reforçam a ideia de que o conhecimento de um *par próprio* (vetor próprio/valor próprio) associado a uma transformação linear é essencial para caracterizar essa transformação. Os métodos matemáticos que exploram as propriedades de uma matriz usando só o seu espectro são conhecidos por métodos espectrais e estão presentes em inúmeras aplicações. Por outro lado, o sistema (1.1) implica que o conjunto de soluções da seguinte equação é exatamente o espectro de  $A$ :

$$\det(A - \lambda I_n) = 0. \quad (1.2)$$

A equação (1.2) é designada por *equação característica* da matriz  $A$ . Pela definição de determinante concluímos que esta equação é polinomial e tem exatamente grau  $n$ . Ora, o Teorema Fundamental da Álgebra estabelece que um polinómio de ordem  $n$ , em  $\mathbb{C}$ , tem exatamente  $n$  raízes (contadas as suas multiplicidades). Essas raízes serão reais ou complexas. Se  $A$  for uma matriz de entradas reais, a equação característica só pode admitir uma raiz complexa se também admitir outra raiz que é a sua conjugada.

Em princípio, haverá duas classes de métodos numéricos para resolver a equação (1.2):

- Métodos Diretos: são métodos que calculam exatamente as soluções, num número finito de passos, desde que os cálculos sejam feitos sem erros de arredondamento;
- Métodos Iterativos: são métodos que definem uma sucessão (de escalares, vetores ou matrizes) convergente para um limite que define as soluções. Mesmo que os cálculos sejam feitos sem erros de arredondamento, as soluções não serão encontradas sem um erro de truncatura, pois não é possível calcular o limite da sucessão definida. Na verdade, os cálculos serão interrompidos quando estiver satisfeita uma condição de paragem, que estabelece normalmente uma tolerância para o erro das soluções.

Uma das principais características do cálculo numérico dos valores próprios de uma matriz é a não existência de métodos diretos para matrizes de ordem superior a quatro. Para  $n = 2$  existe a fórmula resolvente das equações do segundo grau, para  $n = 3$  existem as fórmulas de Cardano e para  $n = 4$  as fórmulas de Ferrari. Estas fórmulas permitem encontrar as soluções num número finito de passos envolvendo as operações da aritmética de números reais (soma, subtração, multiplicação e divisão) e o cálculo de raízes de índice  $n$ ,  $n \in \mathbb{N}$ . Note-se que esta última operação não é classificada como um passo racional porque não pode ser feita num número finito de operações de aritmética. As fórmulas de Cardano e de Ferrari não são muito práticas para cálculo manual mas qualquer sistema simbólico de álgebra computacional (*Symbolic Toolbox do MATLAB®*, *MAPLE®*, *MATHEMATICA®*) permite obter as soluções exatas, nestes casos ( $n = 2, 3, 4$ ). O que pode acontecer é que as expressões algébricas que definem as soluções sejam complicadas, por estarem expressas como frações envolvendo radicais. Esta questão preocupou os matemáticos durante séculos. Na verdade, houve imensas tentativas para resolver a equação

de ordem 5 mas as fórmulas nunca foram encontradas. Deve referir-se que, ao longo dos séculos, houve matemáticos que julgavam ter conseguido deduzir tais fórmulas, mas vinha sempre a revelar-se que se haviam enganado nas deduções. Para piorar (ou melhorar) a referida situação, é bem sabido que já foi demonstrado que não existem métodos diretos para resolver o problema de calcular as raízes de equações polinomiais de grau  $n = 5, 6, 7, \dots$ . Isso foi demonstrado em 1824 por Niels Abel (1802-1829) e constitui o enunciado do conhecido Teorema da Impossibilidade de Abel-Ruffini ([24, Cap. 7, 8, 9, 12 e 13]). Deste modo, torna-se incontornável o uso de métodos numéricos iterativos para o cálculo de valores próprios de matrizes de ordem superior a quatro. Com esses métodos numéricos desistimos de calcular exatamente as raízes da equação caraterística. Na verdade, a essência desses métodos consiste em definir sucessões (escalares, vectoriais ou matriciais) convergentes, cujos limites são definidos em termos de alguns (ou todos) os valores próprios da matriz. Assim, em termos computacionais, partimos de uma aproximação inicial e, recursivamente, calculamos sucessivos termos da sucessão definida. Um critério de paragem pré-fixado permite-nos interromper o cálculo, num dado termo da sucessão, de forma a que se garanta que o erro tolerado não é excedido. Assim, o que se calcula são aproximações dos valores próprios afetadas de erros de truncatura (pelo facto de se ter interrompido os cálculos) e de um eventual erro de arredondamento, se usarmos aritmética de precisão finita. O uso de aritmética computacional de vírgula flutuante torna-se inevitável, para matrizes de grande dimensão, uma vez que a rapidez desta aritmética é muito superior à de qualquer sistema simbólico de álgebra computacional. Note-se que estes sistemas de álgebra computacional permitiriam, hipoteticamente, encontrar aproximações para os valores próprios que não estivessem afetadas de erros de arredondamento. Mas o inevitável erro de truncatura afetaria sempre os resultados. Assim, há toda a vantagem em usar aritmética em vírgula flutuante (em vez de aritmética exata) salvo em casos excecionais.

O objetivo principal desta dissertação é definir e estudar métodos numéricos iterativos de cálculo de valores próprios de matrizes quadradas de entradas reais.

## 1.2 Notas Históricas

Os valores próprios são introduzidos, atualmente, no contexto do estudo da teoria de matrizes. Contudo, historicamente apareceram associados ao estudo de formas quadráticas e equações diferenciais. Leonhard Euler (1707–1781) estudou o movimento rotacional de um corpo rígido e descobriu a importância dos principais eixos do movimento. Lagrange (1736–1813) concluiu que os principais eixos do movimento eram os vetores próprios da matriz de inércia do corpo. No princípio do século XIX, Cauchy aplicou esses resultados para classificar as superfícies quadráticas. Foi Cauchy (1789-1957) que cunhou o termo francês “*racine caractéristique*” para o que, atualmente, chamamos valor próprio. Esse termo sobrevive, no Português, na designação de equação caraterística.

Fourier (1772-1837) usou o trabalho de Lagrange para resolver a equação do calor, no seu famoso trabalho de 1822 ( “*Théorie Analytique de la Chaleur*”). Sturm (1803-1855) e Cauchy provaram que as matrizes reais simétricas tinham valores próprios reais. Em 1855,

Hermite (1822-1901) provou que o mesmo acontecia para matrizes de entradas complexas, a que hoje chamamos hermiticas.

Muitas outras contribuições foram feitas para a teoria dos valores próprios, não só no contexto de espaços de dimensão finita mas também em espaços de dimensão infinita. Referiremos só que, no início do século XX, Hilbert (1862-1943) estudou os valores próprios de operadores integrais, definindo esses operadores como estando associados a matrizes infinitas. Foi Hilbert que usou inicialmente o termo alemão “*eigen*” para designar valores próprios e vetores próprios numa publicação<sup>1</sup> de 1904.

A teoria foi sendo desenvolvida e as aplicações práticas foram surgindo nas mais variadas situações, uma vez que o cálculo de valores próprios ficou ligado à teoria das equações diferenciais e à teoria da estabilidade de sistemas. Contudo, as necessidades computacionais exigidas nas aplicações eram de tal ordem que a simples ideia de construir a solução numérica era absurda. Os métodos inicialmente propostos passavam por fazer transformações de semelhança na matriz dada até encontrar uma matriz companheira que exibisse os coeficientes do polinómio caraterístico. De seguida, chegava-se ao bem conhecido problema de resolver uma equação polinomial. Contudo, a complexidade do problema de calcular raízes de polinómios caraterísticos (i.e., valores próprios de matrizes) aumenta rapidamente com o aumento do grau do polinómio (dimensão da matriz). Para piorar tudo, qualquer procedimento computacional que passasse por calcular primeiro os coeficientes do polinómio caraterístico podia ser muito impreciso na presença dos inevitáveis erros de arredondamento, porque as raízes de um polinómio podem ser muito sensíveis a variações (ainda que pequenas) nos coeficientes.

Note-se que em 1846, Jacobi (1804-1851) propôs um método iterativo matricial para o cálculo dos valores próprios de uma matriz real e simétrica. No entanto, esse método não foi muito usado porque exigia muitos cálculos que, na época, eram feitos manualmente. O método de Jacobi só foi usado extensamente a partir de 1950 com o advento dos computadores digitais. Na verdade, apesar de numerosos esforços para a construção de dispositivos de cálculo, questões tecnológicas limitaram sempre as capacidades desses dispositivos. Nas primeiras décadas do século XX, o cálculo das raízes de um polinómio de grau elevado era uma tarefa quase impossível. No entanto, em 1920, apareceu um método, devido a Leonhard Bairstow, que usava só aritmética de reais e conseguia calcular as raízes reais ou complexas de um polinómio, com ordem de convergência quadrática. Apesar desse desenvolvimento teórico, nas primeiras décadas do século XX, o cálculo das raízes de um polinómio de grau elevado era uma tarefa quase impossível.

Outro algoritmo de cálculo de valores próprios, baseado em recursões matriciais, foi o Método da Potência que apareceu na literatura em 1929, publicado por Richard Edler von Mises (1883–1953). Contudo, só após a segunda guerra mundial os computadores digitais de elevada velocidade de processamento se tornaram disponíveis. Após isso, a procura de algoritmos iterativos adequados fez-se sentir. Na verdade, os matemáticos começaram

---

<sup>1</sup> Na verdade, no início do século XX, os termos usados em inglês eram “*proper value*” e “*proper vector*”. No entanto, como é sabido, atualmente na literatura anglo-saxónica esses termos são, respetivamente, “*eigenvalues*” e “*eigenvectors*.” Estes termos impuseram-se por serem designações muito distintas de qualquer palavra inglesa e, portanto, não geraram confusões com outras noções ou objetos matemáticos.

a propor métodos especialmente dirigidos para o cálculo automático em computadores digitais.

Com o aparecimento do Método QR, em 1961, ficou possível o cálculo eficiente e com elevada precisão de todos os valores próprios de uma matriz, para uma gama muito genérica de matrizes. Por isso, alguns autores ([26]) chamam-lhe “o rei dos métodos”. Este método tem sido estudado e melhorado até aos nossos dias e foi proposto, independentemente, por John G.F. Francis e Vera Kublanovskaya. O método QR tem muitas variantes. No entanto, o uso de matrizes de Householder permite obter a melhor *performance* em termos de eficiência computacional, especialmente se combinadas com a clássica decomposição LU. As transformações de Householder são um poderoso processo de triangularização de matrizes e foram propostas em 1958 por Alston Scott Householder.

Para matrizes hermiticas esparsas, o algoritmo de Lanczos é um dos exemplos de métodos iterativos, entre outras possibilidades. Esse algoritmo foi proposto por Cornelius Lanczos (1893-1974) durante o período 1949–1952 em que trabalhou no *Institute for Numerical Analysis* ligado à Universidade da Califórnia ([12, pp. 894–897], [28], [16] e [17]).

Devemos frisar que, por exemplo, o MATLAB® faz o cálculo das raízes dos polinómios construindo uma matriz companheira. Depois, aplica um método iterativo matricial para calcular os valores próprios dessa matriz. É uma ironia histórica porque isto só se faz nos últimos cinquenta anos. No passado, era o cálculo dos valores próprios que era convertido no velho e bem conhecido problema de calcular raízes de polinómios.

A investigação no área do *Matrix Eigenvalue Problem* continua ativa porque se procuram métodos mais eficientes e precisos. Além disso, há sempre problemas difíceis, como é o caso das matrizes com valores próprios muito próximos uns dos outros. Essas matrizes, ditas de espectro mal separado, constituem um desafio numérico, uma vez que, em geral, os métodos apresentam dificuldades de convergência.

## 1.3 Aplicações Práticas

Como já foi dito, a teoria dos valores próprios nasceu em estudos de mecânica de rotação de corpos rígidos. Na verdade, uma vez que o cálculo de valores próprios ficou ligado à teoria das equações diferenciais as aplicações do cálculo numérico dos valores próprios são imensas, uma vez que os sistemas dinâmicos são dos modelos mais usados nas aplicações. Além disso, um aspeto central da teoria dos sistemas são os estudos de estabilidade. Ora, esses estudos podem ser formulados como problemas de localização de valores próprios (no círculo unitário, por exemplo).

Já no século XX, surgiram aplicações que exigem o cálculo numérico de valores próprios ([28]):

- Mecânica Quântica: no estudo da equação de Schrodinger e na teoria das órbitas de eletrões em átomos e moléculas;
- Geologia e Glaciologia;

- Estatística: na análise de componentes principais (ACP) de grandes bases de dados com aplicações em “*data mining*”, investigação química, Psicologia e Marketing;
- Processamento de Imagem: reconhecimento de padrões e mesmo de rostos humanos;
- Estudo epidemiológico de doenças infecciosas;
- Teoria Espetral de Grafos: um grafo pode ser definido pela sua matriz de adjacência e dessa forma as características do grafo podem ser estabelecidas conhecendo o espectro do grafo;
- Algoritmos de *PageRank*: estes algoritmos estão na base da construção dos motores de busca usados nas pesquisas da internet, como por exemplo o Google®. Na verdade, o vetor próprio dominante de um grafo é usado para medir a centralidade dos seus vértices. Um exemplo é o “*Google® PageRank Algorithm*.” Neste algoritmo, o valor próprio de uma matriz de adjacência modificada (pela busca) do grafo, que é a “*World Wide Web*”, dá o “*page rank*” (ou seja, o número de ordenação de cada site numa busca). Esses números surgem nas componentes do vetor próprio dominante. Assim, o sucesso civilizacional que é o Google® só é possível porque existem métodos de cálculo dos valores próprios dominantes de matrizes de muito grande dimensão, bastante eficientes. Refira-se que a patente deste algoritmo, que está ao serviço exclusivo da *Google®*, rendeu, ao que consta, à Universidade de Stanford mais de 330 milhões de dólares ([18] e [6]).

## 1.4 Objetivos Gerais do Trabalho

Como já foi dito, o objetivo principal desta dissertação é definir e estudar métodos numéricos iterativos de cálculo de valores próprios de matrizes de entradas reais. No entanto, pretendemos também estudar alguns métodos de localização de valores próprios, uma vez que estes métodos são muito úteis para obter aproximações iniciais necessárias aos métodos iterativos. Nos métodos de localização estão incluídos métodos que permitem *separar* os valores próprios em subdomínios disjuntos.

Optámos por abordar várias classes de métodos que têm importância histórica para ilustrar as diferentes maneiras em que podem ser usadas, para abordar o problema. Na verdade, há duas grandes classes de métodos:

- Métodos de obtenção do polinómio característico, a partir da matriz dada, conjugados com métodos numéricos iterativos de obtenção das raízes desse polinómio;
- Métodos iterativos matriciais, que são os mais usados atualmente.

Dada a vastidão do tema abordado houve que fazer opções e escolhas para limitar a extensão desta dissertação. Por um lado, não é possível descrever os métodos todos. Por outro lado, traçámos como objetivo o desenvolvimento do trabalho numa vertente mais



ligada à teoria, que sustenta a consistência e a convergência dos métodos definidos. Essa escolha levou a que as análises de erro (de truncatura e de arredondamento) não fossem feitas. Da mesma forma, não fizemos estudos acerca do condicionamento do problema, nem análises de estabilidade numérica dos diferentes métodos. Convém frisar que os tópicos que não foram tratados só poderiam ser abordados após o trabalho desta dissertação e alguns deles são, ainda hoje, matéria de investigação.

## 1.5 Organização da Dissertação

No Capítulo 2, começaremos por introduzir algumas noções preliminares que auxiliarão no desenvolvimento do tema, como por exemplo, normas matriciais e transformações de semelhança.

Dedicamos o Capítulo 3 à localização de valores próprios a partir da matriz (sem recurso ao polinómio caraterístico). Abordamos métodos baseados no cálculo de normas e em círculos de Gershgorin. No Apêndice A são apresentados métodos de localização clássicos baseados nos coeficientes do polinómio caraterístico.

No Capítulo 4 estudaremos quatro modos diferentes de obter o polinómio caraterístico de uma dada matriz, o método de Danilevsky, o método de Krylov, o método de Leverrier e o método dos coeficientes indeterminados. Como estes métodos só resolvem o subproblema de converter o cálculo dos valores próprios no cálculo das raízes de um polinómio, dedicamos algum espaço (Capítulo 5) a métodos iterativos especialmente aplicados a aproximar zeros de polinómios. Os métodos apresentados serão o método de Newton-Horner, que podemos ver como uma adaptação melhorada ao método de Newton-Raphson, o método de Müller, com uma ideia base idêntica ao método da secante, e o método de Bairstow, que extrai de um polinómio um fator quadrático.

Chegamos então à importante classe dos Métodos Matriciais Iterativos de Aproximação de Valores Próprios. Esse assunto é o tema do Capítulo 6. Dentro de uma vasta gama de métodos que poderiam ser abordados, optámos por desenvolver o Método das Potências, que devolve uma aproximação do valor próprio de maior valor absoluto, o Método QR, que devolve uma aproximação de todo o espetro de uma matriz, e o Método de Jacobi, que devolve todo o espetro de uma matriz real e simétrica, se determinadas condições de convergência forem satisfeitas.

No Apêndice B incluímos o código MATLAB<sup>®</sup> das rotinas que foram programadas para testar os métodos e fazer as simulações que conduziram aos exemplos numéricos que atravessam o texto.



# Capítulo 2

## Preliminares

Neste capítulo, iremos apresentar algumas definições e resultados importantes que servirão de apoio ao desenvolvimento dos capítulos seguintes. Começaremos pelas normas matriciais, passando por noções básicas acerca de valores e vetores próprios e pela semelhança de matrizes. Por fim, definiremos matrizes de Householder e matrizes de Givens.

### 2.1 Normas matriciais

Sendo as normas matriciais um tema clássico é possível encontrá-lo exposto em diversa literatura, por exemplo, em [20, pp. 18–27] ou [22, cap. 4]. Neste trabalho, optámos por seguir a notação de [15, cap. 10] e [10, cap. 5].

Antes de definirmos norma matricial, começaremos por definir norma vetorial.

**Definição 2.1** *Consideremos  $V$  um espaço vetorial sobre  $\mathbb{C}$ . A  $\|\cdot\|$  chamamos **norma vetorial** se, para todo  $x, y \in V$ , se verificarem as seguintes propriedades:*

1.  $\|x\| \geq 0$ ;
2.  $\|x\| = 0 \Leftrightarrow x = 0$ ;
3.  $\|\alpha x\| = |\alpha| \|x\|, \forall \alpha \in \mathbb{C}$ ;
4.  $\|x + y\| \leq \|x\| + \|y\|$  (*desigualdade triangular*).

Apresentamos alguns exemplos de normas vetoriais que se podem definir no espaço vetorial  $\mathbb{C}^n$  e que são, também, as mais frequentemente usadas.

- $\|x\|_2 = \sqrt{\sum_{i=1}^n |x_i|^2}$ , (norma euclidiana);
- $\|x\|_1 = \sum_{i=1}^n |x_i|$ , (norma absoluta);
- $\|x\|_\infty = \max_{1 \leq i \leq n} (\sum_{i=1}^n |x_i|)$ , (norma do máximo);

- $\|x\|_p = (\sum_{i=1}^n |x_i|^p)^{\frac{1}{p}}$ , com  $p \geq 1$ , (normas  $l_p$  ou normas  $p$  de Holder).

**Definição 2.2** Uma norma vetorial  $|||\cdot|||$  em  $\mathbb{C}^{n \times n}$  diz-se **norma matricial** se:

$$\forall A, B \in \mathbb{C}^{n \times n} \quad |||AB||| \leq |||A||| \quad |||B|||. \quad (2.1)$$

Seja  $A = [a_{ij}]$ ,  $i, j = 1, \dots, n$ , uma matriz de entradas complexas, as seguintes normas vetoriais são também normas matriciais:

- $\|A\|_1 = \sum_{i,j=1}^n |a_{ij}|$ ;
- $\|A\|_F = \sqrt{\sum_{i,j=1}^n |a_{ij}|^2}$ , (norma euclidiana ou de Frobenius);
- $\|A\|_\infty = \max_{1 \leq i,j \leq n} |a_{ij}|$ .

Pode-se consultar a verificação das condições da definição de norma matricial, para estes casos, por exemplo em [10, pp. 291–292].

Algumas propriedades verificadas pelas normas matriciais são as seguintes:

**Propriedade 2.1** Seja  $A \in \mathbb{C}^{n \times n}$ , então:

1.  $|||I_n||| \geq 1$ , onde  $I_n$  é a matriz identidade de ordem  $n$ ;
2.  $|||A^k||| \leq |||A|||^k$ , qualquer que seja  $k \in \mathbb{Z}$ ;
3.  $|||A^{-1}||| \geq \frac{1}{|||A|||}$ , com  $\det(A) \neq 0$ .

**Proposição 2.1** Se  $|||\cdot|||$  é uma norma matricial e se  $S \in \mathbb{C}^{n \times n}$  é uma matriz invertível então

$$|||A|||_S = |||S^{-1}AS|||$$

é uma norma matricial.

**Prova:** A prova desta proposição facilmente decorre do facto de  $|||\cdot|||$  ser uma norma matricial e para que  $|||\cdot|||_S$  seja uma norma matricial terá de verificar as Definições 2.1 e 2.2.

Sejam  $A$  e  $B$  matrizes quadradas com entradas em  $\mathbb{C}$ ,  $|||\cdot|||$  uma norma matricial e  $S$  uma matriz invertível.

$$|||A|||_S = |||S^{-1}AS||| \geq 0;$$

$$|||A|||_S = 0 \Leftrightarrow |||S^{-1}AS||| = 0 \Leftrightarrow S^{-1}AS = 0 \Leftrightarrow A = 0;$$

$$|||\alpha A|||_S = |||\alpha(S^{-1}AS)||| = |\alpha| |||S^{-1}AS||| = |\alpha| |||A|||_S, \forall \alpha \in \mathbb{C};$$

$$\begin{aligned}
|||A + B|||_S &= |||S^{-1}(A + B)S||| \\
&= |||(S^{-1}AS) + (S^{-1}BS)||| \\
&\leq |||S^{-1}AS||| + |||S^{-1}BS||| \\
&= |||A|||_S + |||B|||_S; \\
\\
|||AB|||_S &= |||S^{-1}(AB)S||| \\
&= |||(S^{-1}AS)(S^{-1}BS)||| \\
&\leq |||S^{-1}AS||| |||S^{-1}BS||| \\
&= |||A|||_S |||B|||_S.
\end{aligned}$$

□

Uma outra noção muito importante é a de norma matricial induzida por uma norma vetorial, que introduzimos de seguida.

**Definição 2.3** *Sejam  $A$  uma matriz quadrada de ordem  $n$ ,  $x$  um vetor não nulo de  $\mathbb{C}^n$  e  $\|\cdot\|$  uma norma vetorial, a*

$$|||A||| = \sup_{\substack{x \in \mathbb{C}^n \\ x \neq 0}} \frac{\|Ax\|}{\|x\|}$$

*chamamos **norma matricial induzida** pela norma vetorial  $\|\cdot\|$ .*

Para normas matriciais induzidas verificam-se as seguintes propriedades:

**Propriedade 2.2** *Sejam  $A$  uma matriz quadrada de ordem  $n$ ,  $x$  um vetor não nulo de  $\mathbb{C}^n$ ,  $\|\cdot\|$  uma norma vetorial e  $|||\cdot|||$  uma norma matricial induzida por  $\|\cdot\|$ . Verifica-se que:*

1.  $|||I_n||| = 1$ ;
2.  $\|Ax\| \leq |||A||| \|x\|$ .

Note-se que quando uma norma matricial e uma norma vetorial verificam o ponto 2 da Propriedade 2.2 estas dizem-se **compatíveis**.

As normas matriciais induzidas mais usuais são as seguintes:

- $|||A|||_1 = \max_{1 \leq j \leq n} (\sum_{i=1}^n |a_{ij}|)$ , (máximo da soma, em valor absoluto, por colunas, norma induzida pela norma  $\|\cdot\|_1$ );
- $|||A|||_2 = \sqrt{\rho(A^*A)}$ , onde  $\rho(B)$  denota o raio espectral da matriz  $B$ , que definiremos mais adiante, (norma espectral, induzida pela norma euclidiana);
- $|||A|||_\infty = \max_{1 \leq i \leq n} (\sum_{j=1}^n |a_{ij}|)$ , (máximo da soma, em valor absoluto, por linhas, norma induzida pela norma  $\|\cdot\|_\infty$ ).

Tendo em atenção a definição de transconjugada de uma matriz podemos ver que  $|||A|||_1 = |||A^*|||_\infty$  ou que  $|||A^*|||_1 = |||A|||_\infty$ .

## 2.2 Valores e Vetores Próprios de uma matriz

**Definição 2.4** *Seja  $A \in \mathbb{C}^{n \times n}$  uma matriz. Dizemos que  $\lambda \in \mathbb{C}$  é um **valor próprio** de  $A$  se existir  $x \in \mathbb{C}^n$ , não nulo, tal que:*

$$Ax = \lambda x. \quad (2.2)$$

*A  $x$  chamamos **vetor próprio** de  $A$  **associado a**  $\lambda$ .*

A equação matricial (2.2) pode ser escrita na seguinte forma:

$$(\lambda I_n - A)x = 0. \quad (2.3)$$

Assim,  $\lambda$  é valor próprio de  $A$  se e só se a matriz  $(\lambda I_n - A)$  for não invertível, isto é,  $\lambda$  é valor próprio de  $A$  se e só se  $\det(\lambda I_n - A) = 0$ . Ainda podemos dizer que  $\lambda$  é valor próprio de  $A$  se e só se  $\lambda$  é raiz do polinómio  $\det(\lambda I_n - A)$ .

**Definição 2.5** *Ao polinómio*

$$p_A(\lambda) = \det(\lambda I_n - A) \quad (2.4)$$

*chamamos **polinómio caraterístico da matriz**  $A$ .*

Se  $p_A(\lambda)$  é o polinómio caraterístico de  $A$  então:

$$p_A(\lambda) = (\lambda - \lambda_1)^{m_1} \cdots (\lambda - \lambda_k)^{m_k},$$

com  $m_1 + \cdots + m_k = n$ , onde  $\lambda_1, \dots, \lambda_k$  são os valores próprios distintos de  $A$ .

A  $m_i$  chamamos **multiplicidade algébrica** do valor próprio  $\lambda_i$ , para  $i = 1, \dots, k$ , e escrevemos  $\alpha_A(\lambda_i) = m_i$ .

**Definição 2.6** *Ao conjunto dos valores próprios da matriz  $A$  damos o nome de **espectro de**  $A$  e representamos por  $\sigma(A)$ , isto é,*

$$\sigma(A) = \{\lambda \in \mathbb{C} : \lambda \text{ é valor próprio de } A\}. \quad (2.5)$$

**Definição 2.7** *Designamos por **raio espectral de**  $A$ , denotado por  $\rho(A)$ , o valor*

$$\rho(A) = \max \{|\lambda|, \lambda \in \sigma(A)\}. \quad (2.6)$$

**Definição 2.8** *Se  $A$  for uma matriz quadrada de ordem  $n$  e  $\lambda$  for um seu valor próprio, ao conjunto formado pelos vetores próprios associados a  $\lambda$  e pelo vetor nulo chamamos **subespaço próprio de**  $A$  **associado ao valor próprio**  $\lambda$  e denotamos por  $S_A(\lambda)$ .*

A  $\gamma_A(\lambda) = \dim(S_A(\lambda))$  damos o nome de **multiplicidade geométrica** do valor próprio  $\lambda$ .

**Definição 2.9** *Seja  $\lambda$  um valor próprio da matriz  $A$  e  $x$  um vetor próprio associado a  $\lambda$ . Ao par  $(\lambda, x)$  chamamos **par próprio**. Se  $|\lambda| = \rho(A)$ , ao par  $(\lambda, x)$  chamamos **par próprio dominante** e a  $\lambda$  **valor próprio dominante**.*

Apresentamos, de seguida, algumas propriedades dos valores próprios de uma dada matriz  $A$ .

**Propriedade 2.3** *Se  $\lambda$  é valor próprio de  $A$  então:*

1.  $\lambda$  é valor próprio de  $A^*$ ;
2.  $\alpha\lambda$  é valor próprio de  $\alpha A$ ,  $\forall \alpha \in \mathbb{C}$ ;
3.  $\lambda^k$  é valor próprio de  $A^k$ , com  $k \geq 0$  inteiro;
4.  $\lambda^{-1}$  é valor próprio de  $A^{-1}$ , com  $A$  invertível;

As provas destas propriedades podem ser vistas em [1, pp.483–486].

A propriedade seguinte também é importante e a sua prova pode ser vista em [1, p. 490]

**Propriedade 2.4** *Se  $A \in \mathbb{R}^{n \times n}$  for uma matriz simétrica então os seus valores próprios são reais.*

Seguidamente, enunciamos um teorema que revela a relação existente entre os valores próprios de uma matriz e o traço<sup>1</sup> dessa mesma matriz. A prova deste teorema pode ser vista em [15, p. 157]. Denotaremos por  $\text{tr}(A)$  o traço de uma dada matriz  $A$ .

**Teorema 2.1** *Seja a matriz  $A \in \mathbb{C}^{n \times n}$  e sejam  $\lambda_1, \dots, \lambda_k$  os seus valores próprios distintos. Então:*

$$\text{tr}(A) = \sum_{i=1}^k \alpha_A(\lambda_i) \lambda_i. \quad (2.7)$$

Para este trabalho também será útil a noção de matriz companheira da matriz  $A$ .

**Definição 2.10** *Seja  $p(\lambda) = \lambda^n + p_{n-1}\lambda^{n-1} + \dots + p_1\lambda + p_0$  o polinómio caraterístico de uma dada matriz  $A$ . A matriz*

$$C_A = \begin{bmatrix} -p_{n-1} & -p_{n-2} & \cdots & -p_1 & -p_0 \\ 1 & 0 & \cdots & 0 & 0 \\ 0 & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 1 & 0 \end{bmatrix} \quad (2.8)$$

*chamamos matriz companheira do polinómio caraterístico de  $A$  ou, simplesmente, matriz companheira de  $A$ .*

---

<sup>1</sup>O traço de uma matriz é dado pela soma dos elementos da sua diagonal principal.

A matriz companheira também pode ser dada nas seguintes formas:

$$\begin{bmatrix} -p_{n-1} & 1 & \cdots & 0 & 0 \\ -p_{n-2} & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & 1 & 0 \\ -p_1 & 0 & \cdots & 0 & 1 \\ -p_0 & 0 & \cdots & 0 & 0 \end{bmatrix} \quad \text{ou} \quad \begin{bmatrix} 0 & 0 & \cdots & 0 & -p_0 \\ 1 & 0 & \cdots & 0 & -p_1 \\ 0 & 1 & \cdots & 0 & \vdots \\ \vdots & \vdots & \ddots & \vdots & -p_{n-2} \\ 0 & 0 & \cdots & 1 & -p_{n-1} \end{bmatrix}$$

ou ainda

$$\begin{bmatrix} 0 & 1 & \cdots & 0 & 0 \\ 0 & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & 1 & 0 \\ 0 & 0 & \cdots & 0 & 1 \\ -p_0 & -p_1 & \cdots & -p_{n-2} & -p_{n-1} \end{bmatrix}.$$

Ao longo do trabalho, quando nos referirmos à matriz companheira da matriz dada estaremos a falar na forma (2.8). Claro que todas as conclusões que se tirem acerca desta forma companheira poderão ser adaptadas para qualquer uma das outras. De referir, ainda, que a matriz companheira partilha, com a matriz original, o polinómio caraterístico.

## 2.3 Transformações de Semelhança

Algumas transformações de semelhança de matrizes revelam-se muito importantes no desenrolar de alguns métodos de aproximação de valores próprios, que irão ser abordados em capítulos posteriores. Por isso, apresentamos, nesta secção, alguns resultados que iremos utilizar mais adiante.

**Definição 2.11** *As matrizes  $A$  e  $B$ , quadradas de ordem  $n$ , dizem-se **matrizes semelhantes** se existe uma matriz invertível  $S \in \mathbb{C}^{n \times n}$  tal que:*

$$A = S^{-1}BS.$$

*À passagem de  $B$  para  $S^{-1}BS$  chamamos **transformação de semelhança**. À matriz  $S$  chamamos **matriz de semelhança**.*

**Teorema 2.2** *Sejam  $A$  e  $B$  matrizes quadradas de ordem  $n$ , semelhantes. Então  $A$  e  $B$  possuem os mesmos valores próprios. Por outras palavras, transformações de semelhança preservam o polinómio caraterístico.*

A prova deste teorema é relativamente simples e, para leitores mais interessados, pode ser vista em [1, pp. 497–498]. Relembremos que o recíproco deste Teorema é falso. Um exemplo muito simples é o seguinte, tomemos as matrizes  $A$  e  $B$ , tais que,

$$A = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}$$



e

$$B = I_2 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}.$$

O polinómio caraterístico destas duas matrizes é igual, ou seja,

$$p_A(x) = p_B(x) = x^2 - 2x + 1.$$

Se as matrizes  $A$  e  $B$  fossem semelhantes existiria uma matriz  $S$  invertível tal que  $A = S^{-1}BS$ , donde

$$A = S^{-1}I_2S = S^{-1}S = I_2,$$

mas  $A \neq I_2$ . Logo não podemos afirmar que, se duas matrizes possuem o mesmo polinómio caraterístico então são semelhantes.

**Definição 2.12** *Seja  $A$  uma matriz quadrada de ordem  $n$ . A diz-se **diagonalizável** se é semelhante a uma matriz diagonal.*

As matrizes unitárias e as transformações de semelhança que envolvem estas matrizes, introduzidas a seguir, serão de grande importância no desenrolar do último capítulo deste trabalho. Assim, começamos por definir matrizes unitárias.

**Definição 2.13** *Uma matriz  $U \in \mathbb{C}^{n \times n}$  diz-se **unitária** se  $U^*U = I_n$ . No caso de  $U$  ser real temos  $U^T U = I_n$  e dizemos que  $U$  é **ortogonal**.*

Em [10, pp. 67–68], encontramos um teorema que revela algumas propriedades acerca das matrizes unitárias e onde também se pode ver a demonstração de tais propriedades. Aqui, enunciamos algumas que iremos utilizar.

**Teorema 2.3** *Seja a matriz  $U \in \mathbb{C}^{n \times n}$ , as seguintes proposições são equivalentes:*

1.  $U$  é unitária
2.  $U$  é não singular e  $U^* = U^{-1}$
3.  $UU^* = I_n$

A propriedade seguinte também será muito útil ao longo do Capítulo 6 e facilmente se verifica.

**Propriedade 2.5** *Sejam  $U, V \in \mathbb{C}^{n \times n}$ . Se  $U$  e  $V$  são duas matrizes unitárias então a matriz  $UV$  é também unitária.*

**Definição 2.14** *Uma matriz  $B \in \mathbb{C}^{n \times n}$  diz-se **unitariamente semelhante** a  $A \in \mathbb{C}^{n \times n}$  se existe uma matriz unitária  $U \in \mathbb{C}^{n \times n}$  tal que*

$$B = U^*AU.$$

*Caso a matriz  $U$  seja real dizemos que  $B$  é **ortogonalmente semelhante** a  $A$ .*

As matrizes de forma triangular são mais simples, no que diz respeito à determinação dos seus valores próprios, pois estes são as entradas da sua diagonal principal. O teorema seguinte garante que existe uma matriz unitária que transforma por semelhança qualquer matriz dada numa triangular superior.

**Teorema 2.4 (Teorema da decomposição de Schur)** *Dada uma matriz  $A \in \mathbb{C}^{n \times n}$  existe uma matriz unitária  $U \in \mathbb{C}^{n \times n}$  e uma matriz triangular superior  $T$  tal que*

$$T = U^*AU.$$

A demonstração deste teorema é feita por indução sobre  $n$  usando o processo de ortogonalização de Gram-Schmidt e um processo de deflação de matrizes. Pode ser visto, por exemplo, em [10, pp. 79–80].

No entanto, nem sempre é fácil encontrar uma matriz unitária  $U \in \mathbb{C}^{n \times n}$  que transforme por semelhança uma dada matriz  $A \in \mathbb{C}^{n \times n}$  numa matriz triangular superior. Note-se que encontrar tal matriz é equivalente a determinar uma base ortonormada de vetores próprios de  $A$  que, se fosse conhecida, tornaria trivial o problema de cálculo dos valores próprios. Deste modo, a redução de uma matriz  $A$  a uma matriz triangular, usando o Teorema da decomposição de Schur, com o objetivo de determinar os seus valores próprios não tem utilidade prática pois, em geral, não é possível determinar, num número finito de passos, uma tal matriz  $U$ . No entanto, como iremos ver no Capítulo 6, na Subsecção 6.2.3, é possível encontrar, num número finito de passos, uma matriz ortogonal  $P$  tal que  $P^TAP$  é uma matriz de Hessenberg. Uma matriz de Hessenberg tem uma forma “quase” triangular e estas matrizes tornam o cálculo de valores próprios mais fácil.

**Definição 2.15** À matriz  $H = [h_{ij}] \in \mathbb{C}^{n \times n}$  chamamos **matriz de Hessenberg** se  $h_{ij} = 0$ , para  $i > j + 1$ , ou seja,

$$H = \begin{bmatrix} h_{11} & h_{12} & \cdots & \cdots & \cdots & h_{1n} \\ h_{21} & h_{22} & \cdots & \cdots & \cdots & h_{2n} \\ 0 & h_{32} & \ddots & & & \vdots \\ 0 & 0 & \ddots & \ddots & & \vdots \\ \vdots & \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 & h_{n,n-1} & h_{nn} \end{bmatrix}.$$

**Exemplo 2.1** A matriz

$$H = \begin{bmatrix} 1 & 2 & 4 & 5 & 0 \\ 1 & 0 & 1 & 3 & 2 \\ 0 & 2 & 0 & 7 & 4 \\ 0 & 0 & 2 & 1 & 1 \\ 0 & 0 & 0 & 8 & 2 \end{bmatrix}$$

é de Hessenberg ou, podemos dizer que, está na forma superior de Hessenberg.

## 2.4 Matrizes de Householder

Como já referimos, uma matriz dada  $A \in \mathbb{R}^{n \times n}$  pode ser transformada numa matriz triangular ou numa matriz de Hessenberg, por meio de transformações de semelhança. Uma dessas transformações faz uso das matrizes de reflexão de Householder. Além disso, um dos métodos usados para obter a decomposição QR de uma matriz faz uso deste tipo de matrizes, como veremos na Subsecção 6.2.1.

**Definição 2.16** Dado um vetor  $v \in \mathbb{R}^n$  não nulo, arbitrário, **a matriz de reflexão de Householder**, ou simplesmente, **a matriz de Householder**  $P \in \mathbb{R}^{n \times n}$  é dada por

$$P = I_n - 2 \frac{vv^T}{\|v\|_2^2}.$$

Ao vetor  $v$  chamamos **vetor de Householder**.

Podemos verificar que a matriz  $P$  é simétrica,

$$P^T = \left( I_n - 2 \frac{vv^T}{\|v\|_2^2} \right)^T = I_n^T - 2 \frac{(vv^T)^T}{\|v\|_2^2} = I_n - 2 \frac{vv^T}{\|v\|_2^2} = P,$$

e ortogonal,

$$\begin{aligned} P^T P &= P^2 \\ &= \left( I_n - 2 \frac{vv^T}{\|v\|_2^2} \right) \left( I_n - 2 \frac{vv^T}{\|v\|_2^2} \right) \\ &= I_n - 2 \frac{vv^T}{\|v\|_2^2} - 2 \frac{vv^T}{\|v\|_2^2} + 4 \frac{v(v^T v)v^T}{\|v\|_2^4} \\ &= I_n - 4 \frac{vv^T}{\|v\|_2^2} + 4 \frac{vv^T}{\|v\|_2^2} \\ &= I_n. \end{aligned}$$

Dado um vetor  $x \in \mathbb{R}^n$ , o vetor  $y = Px$ , com  $P$  uma matriz de Householder, é uma reflexão de  $x$  em relação a um plano ortogonal ao vetor  $v$ .

**Exemplo 2.2** Consideremos o vetor  $v = \begin{bmatrix} 1 \\ 3 \end{bmatrix}$ . A matriz de Householder é dada por

$$P = I_2 - 2 \frac{vv^T}{\|v\|_2^2} = \frac{1}{5} \begin{bmatrix} 4 & -3 \\ -3 & -4 \end{bmatrix}.$$

Tomemos o vetor  $x = \begin{bmatrix} -2 \\ -2 \end{bmatrix}$ , então  $y = Px = \begin{bmatrix} -\frac{2}{5} \\ \frac{14}{5} \end{bmatrix}$ .

Veja-se a ilustração gráfica da Figura 2.1.

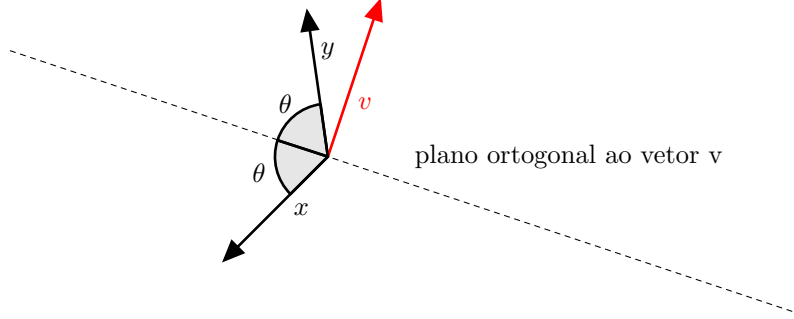


Figura 2.1: Reflexão do vetor  $x$  em relação a um plano ortogonal ao vetor  $v$ .

---

## 2.5 Matrizes de Givens

Assim como as matrizes de Householder, as matrizes de Givens também podem ser utilizadas para se obter a decomposição QR de uma dada matriz, como iremos ver mais adiante. Também são usadas para transformar por semelhança uma matriz dada numa matriz de Hessenberg, como veremos na Subsecção 6.2.3. Estas matrizes são matrizes ortogonais de rotação que irão permitir anular elementos de um vetor ou de uma matriz. Começamos por definir  $(p, q)$ -plano em  $\mathbb{R}^n$ .

**Definição 2.17** Chamamos  $(p, q)$ -**plano em**  $\mathbb{R}^n$ , e denotamos por  $\mathcal{P}_{p,q}$ , com  $1 \leq p < q \leq n$ , ao subespaço de  $\mathbb{R}^n$  gerado por

$$e_p = (0, \dots, 0, \underset{\substack{\uparrow \\ p}}{1}, 0, \dots, 0)$$

$$e_q = (0, \dots, 0, \underset{\substack{\uparrow \\ q}}{1}, 0, \dots, 0).$$

**Definição 2.18** Dados  $p < q$  e um ângulo  $\theta$ , os elementos da **matriz de Givens**,  $G(p, q, \theta)$ , a eles associados são

$$\begin{aligned} g_{pp} = g_{qq} &= \cos(\theta) \\ g_{pq} &= -\sin(\theta) \\ g_{qp} &= \sin(\theta) \\ g_{ii} &= 1 \text{ se } i \neq p, q \end{aligned}$$

e os restantes elementos são todos nulos, ou seja,

$$G \equiv G(p, q, \theta) = \begin{bmatrix} I_{p-1} & \vdots & 0 & \vdots & 0 \\ \cdots & \cos(\theta) & \cdots & -\sin(\theta) & \cdots \\ 0 & \vdots & I_{q-p-1} & \vdots & 0 \\ \cdots & \sin(\theta) & \cdots & \cos(\theta) & \cdots \\ 0 & \vdots & 0 & \vdots & I_{n-q} \end{bmatrix}.$$

Note-se que uma matriz de Givens é uma matriz ortogonal pois, como facilmente se constata,  $G^T G = I_n$ .

As matrizes de Givens são matrizes de rotação. De facto, dado um vetor  $x \in \mathcal{P}_{p,q}$  o vetor  $y = Gx$  é o vetor resultante da rotação, associada ao ângulo  $\theta$  e centrada na origem, do vetor  $x$ .

**Exemplo 2.3** *Tomemos  $n = 3$ , o plano*

$$\mathcal{P}_{1,3} = \left\langle \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \right\rangle$$

e  $\theta = \frac{\pi}{2}$ .

Então, segundo a Definição 2.18,

$$\begin{aligned} G &= \begin{bmatrix} \cos\left(\frac{\pi}{2}\right) & 0 & -\sin\left(\frac{\pi}{2}\right) \\ 0 & 1 & 0 \\ \sin\left(\frac{\pi}{2}\right) & 0 & \cos\left(\frac{\pi}{2}\right) \end{bmatrix} \\ &= \begin{bmatrix} 0 & 0 & -1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix}. \end{aligned}$$

Seja  $v = \begin{bmatrix} 1 \\ 0 \\ 3 \end{bmatrix}$ , aplicando a matriz  $G$  temos

$$\begin{aligned} Gv &= \begin{bmatrix} 0 & 0 & -1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 3 \end{bmatrix} \\ &= \begin{bmatrix} -3 \\ 0 \\ 1 \end{bmatrix}. \end{aligned}$$

Como podemos observar,  $v \in \mathcal{P}_{1,3}$  e  $Gv \in \mathcal{P}_{1,3}$ . Veja-se a ilustração gráfica na Figura 2.2.

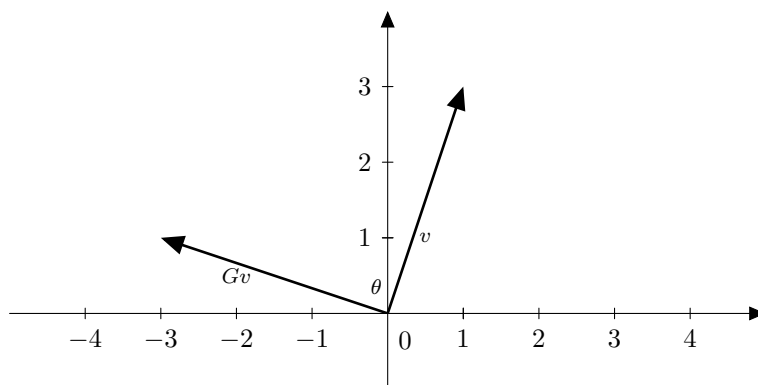


Figura 2.2: Rotação de  $\frac{\pi}{2}$  em torno da origem, do vetor  $v$

---

Para terminar, observe-se que se  $x = (x_1, \dots, x_n)$  então as coordenadas do vetor  $y = Gx$  são dadas por

$$y_j = \begin{cases} x_j & , j \neq p, q \\ \cos(\theta)x_p - \sin(\theta)x_q & , j = p \\ \sin(\theta)x_p + \cos(\theta)x_q & , j = q \end{cases} . \quad (2.9)$$

# Capítulo 3

## Localização de valores próprios

Como se sabe, o problema de calcular os valores próprios de uma matriz pode passar pela determinação das raízes do polinómio caraterístico. No entanto, se esse polinómio for de grau superior a quatro<sup>1</sup> temos de usar métodos numéricos de aproximação de raízes. Esses métodos serão abordados mais à frente, neste trabalho. Antes de se aplicar um destes métodos é necessário obter uma aproximação inicial de uma raiz e, por isso, a sua localização e/ou separação é importante.

Neste capítulo, estudaremos a localização de valores próprios sem recurso ao polinómio caraterístico para, mais tarde, se aplicar no cálculo da sua aproximação. Abordaremos algumas formas de localização de valores próprios usando, para isso, a noção de norma e os círculos de Gershgorin.

A localização de valores próprios com recurso ao polinómio caraterístico pode ser vista no Apêndice A.

### 3.1 Com recurso a normas

Os resultados apresentados, nesta secção, são baseados em [15, cap. 10] e [10, cap. 5].

**Teorema 3.1** *Sejam  $A \in \mathbb{C}^{n \times n}$  uma matriz e  $||| \cdot |||$  uma norma matricial. Para qualquer matriz  $A$  temos que:*

$$\rho(A) \leq |||A|||, \quad (3.1)$$

e

$$\rho(A)^2 \leq |||A|||_1 |||A|||_\infty. \quad (3.2)$$

**Prova:** Seja  $\lambda_0$  um valor próprio de  $A$  tal que  $\rho(A) = |\lambda_0|$ . Então, por definição, existe um vetor  $x \in \mathbb{C}$ , não nulo, tal que  $Ax = \lambda_0 x$ .

---

<sup>1</sup>Para equações de grau superior a 4 existem fórmulas fechadas que nos permitem calcular as suas raízes. Como já referimos, o matemático Abel conseguiu provar que não existem tais fórmulas para equações de grau superior ou igual a 5 ([24, Cap. 7, 8, 9, 12 e 13]).

Tomemos a matriz  $n \times n$  definida da seguinte forma

$$X = \begin{bmatrix} x & 0 & 0 & \cdots & 0 \end{bmatrix}.$$

Podemos ver que

$$\begin{aligned} AX &= A \begin{bmatrix} x & 0 & 0 & \cdots & 0 \end{bmatrix} \\ &= \begin{bmatrix} Ax & 0 & 0 & \cdots & 0 \end{bmatrix} \\ &= \begin{bmatrix} \lambda_0 x & 0 & 0 & \cdots & 0 \end{bmatrix} \\ &= \lambda_0 \begin{bmatrix} x & 0 & 0 & \cdots & 0 \end{bmatrix} \\ &= \lambda_0 X. \end{aligned}$$

Deste modo, temos

$$\|\lambda_0 X\| = \|AX\| \leq \|A\| \|X\|.$$

Como  $x \neq 0$  então  $X \neq 0$  e, portanto,  $\|X\| > 0$ . Assim,

$$|\lambda_0| \leq \|A\|,$$

ou seja,

$$\rho(A) \leq \|A\|,$$

ficando provada a equação (3.1).

Para mostrar (3.2) baseamo-nos em (3.1) e na definição de norma espectral. Assim, como (3.1) é válido para qualquer norma, é válido, em particular, para a norma espectral, logo

$$\rho(A) \leq \|A\|_2 = \sqrt{\rho(A^*A)}$$

donde

$$(\rho(A))^2 \leq \|A\|_2^2 = \rho(A^*A).$$

Então

$$\begin{aligned} (\rho(A))^2 &\leq \rho(A^*A) \\ &\leq \|A^*A\|_1 \\ &\leq \|A^*\|_1 \|A\|_1 = \|A\|_\infty \|A\|_1. \end{aligned}$$

□

**Definição 3.1** O *limite inferior* de uma matriz  $A$ , relativamente a uma norma vetorial  $\|\cdot\|$ , é dado por:

$$\ell(A) = \inf_{\substack{x \in \mathbb{C}^n \\ x \neq 0}} \frac{\|Ax\|}{\|x\|}. \quad (3.3)$$

Esta definição leva-nos aos seguintes resultados:



**Proposição 3.1** Se  $\|\cdot\|$  é uma matriz induzida pela norma vetorial  $\|\cdot\|$  então

$$\ell(A) = \begin{cases} \frac{1}{\|A^{-1}\|} & \text{se } \det(A) \neq 0 \\ 0 & \text{se } \det(A) = 0 \end{cases}.$$

**Prova:** Suponhamos que  $\det(A) \neq 0$ . Então a matriz é invertível.

Por definição,

$$\ell(A) = \inf_{\substack{x \in \mathbb{C}^n \\ x \neq 0}} \frac{\|Ax\|}{\|x\|}.$$

Fazendo a seguinte mudança de variável,  $x = A^{-1}y$ , podemos escrever

$$\ell(A) = \inf_{\substack{y \in \mathbb{C}^n \\ y \neq 0}} \frac{\|y\|}{\|A^{-1}y\|}$$

e, portanto,

$$\begin{aligned} \ell(A) &= \left( \sup_{\substack{y \in \mathbb{C}^n \\ y \neq 0}} \frac{\|A^{-1}y\|}{\|y\|} \right)^{-1} \\ &= \frac{1}{\|A^{-1}\|}. \end{aligned}$$

Suponhamos que  $\det(A) = 0$ . Então existe um vetor  $x_0 \in \mathbb{C}^n$ , não nulo, tal que  $Ax_0 = 0$  e  $\|x_0\| = 1$  donde, por definição de limite inferior,  $\ell(A) = 0$ .  $\square$

**Proposição 3.2** Sejam  $\lambda_1, \dots, \lambda_n$  os valores próprios da matriz  $A \in \mathbb{C}^{n \times n}$ . Então:

$$\ell(A) \leq \min_{1 \leq i \leq n} |\lambda_i|$$

**Prova:** Tendo em conta a definição de limite inferior,

$$\begin{aligned} \ell(A) &= \inf_{\substack{x \in \mathbb{C}^n \\ x \neq 0}} \frac{\|Ax\|}{\|x\|} \\ &= \inf_{\substack{x \in \mathbb{C}^n \\ \|x\|=1}} \|Ax\|. \end{aligned}$$

Sejam  $\lambda_1, \dots, \lambda_n$  os valores próprios de  $A$ . Então, para cada  $i = 1, \dots, n$ , existe  $x^{(i)} \in \mathbb{C}^n$ , não nulo, tal que  $Ax^{(i)} = \lambda_i x^{(i)}$ . Considere-se que  $\|x^{(i)}\| = 1$ , assim

$$\|Ax^{(i)}\| = \|\lambda_i x^{(i)}\| = |\lambda_i|. \quad (3.4)$$

Deste modo, a função  $\|Ax\|$  assume os valores  $|\lambda_i|$ ,  $i = 1, \dots, n$ , logo

$$\ell(A) \leq \min_{0 \leq i \leq n} |\lambda_i|.$$

□

A partir do Teorema 3.1 e das Proposições 3.1 e 3.2 podemos definir, no plano complexo, uma região onde os valores próprios de uma qualquer matriz  $A$  vão estar contidos, ou seja, tendo em conta a mesma norma matricial induzida podemos ver que, para qualquer valor próprio  $\lambda$  de  $A$ ,

$$\ell(A) \leq |\lambda| \leq |||A|||. \quad (3.5)$$

Em certas situações a localização “grosseira” do espectro da matriz permite caraterizar certas propriedades das matrizes. Neste caso, incluem-se as matrizes estáveis. Os resultados que se seguem, além de permitirem uma caraterização espectral das matrizes estáveis, fornecerão uma outra forma de obter, teoricamente, o raio espectral de uma matriz, à custa do limite de uma sucessão que envolve as normas das sucessivas potências de  $A$ .

**Lema 3.1** *Sejam  $A \in \mathbb{C}^{n \times n}$  e  $\varepsilon > 0$ . Então existe uma norma matricial  $|||.|||$  tal que*

$$|||A||| \leq \rho(A) + \varepsilon.$$

**Prova:** Recordemos que o Teorema da decomposição de Schur (Teorema 2.4) diz que existe uma matriz unitária  $U$  e uma matriz triangular superior  $T$  tal que  $A = U^*TU$ , em que a diagonal da matriz  $T$  é constituída pelos valores próprios de  $A$ . Seja  $T$  definida da seguinte forma

$$\begin{bmatrix} \lambda_1 & d_{12} & d_{13} & \cdots & \cdots & d_{1n} \\ 0 & \lambda_2 & d_{23} & \cdots & \cdots & d_{2n} \\ 0 & 0 & \lambda_3 & \cdots & \cdots & d_{3n} \\ \vdots & \vdots & \vdots & \ddots & & \vdots \\ 0 & 0 & 0 & & \ddots & d_{n-1,n} \\ 0 & 0 & 0 & \cdots & \cdots & \lambda_n \end{bmatrix}.$$

Tomemos a matriz diagonal  $D_t = \text{diag}(t, t^2, t^3, \dots, t^n)$ , com  $t > 0$ , e faça-se

$$D_t T D_t^{-1} = \begin{bmatrix} \lambda_1 & t^{-1}d_{12} & t^{-2}d_{13} & \cdots & \cdots & t^{-n+1}d_{1n} \\ 0 & \lambda_2 & t^{-1}d_{23} & \cdots & \cdots & t^{-n+2}d_{2n} \\ 0 & 0 & \lambda_3 & \cdots & \cdots & t^{-n+3}d_{3n} \\ \vdots & \vdots & \vdots & \ddots & & \vdots \\ 0 & 0 & 0 & & \ddots & t^{-1}d_{n-1,n} \\ 0 & 0 & 0 & \cdots & \cdots & \lambda_n \end{bmatrix}. \quad (3.6)$$

Para um  $t > 0$ , suficientemente grande, a soma de todos os elementos da matriz, em valor absoluto, exceto os elementos da diagonal, é menor do que  $\varepsilon$ . Em particular,

$$|||D_t T D_t^{-1}|||_1 \leq \rho(A) + \varepsilon.$$

Recorrendo à Proposição 2.1, definamos a norma matricial  $|||.|||$  da seguinte forma

$$|||A||| := |||(U D_t^{-1})^{-1} A (U D_t^{-1})|||_1$$

e, portanto,

$$|||A||| := |||D_t U^* A U D_t^{-1}|||_1 = |||D_t T D_t^{-1}|||_1.$$

Se escolhermos  $t$  suficientemente grande, temos que  $|||A||| \leq \rho(A) + \varepsilon$ , como queríamos.  $\square$

**Lema 3.2** *Seja  $A \in \mathbb{C}^{n \times n}$ . Se existe uma norma matricial  $|||\cdot|||$  tal que  $|||A||| < 1$  então*

$$\lim_{k \rightarrow \infty} A^k = 0, \quad (3.7)$$

*ou seja, todas as entradas de  $A^k$  tendem para zero quando  $k$  tende para  $\infty$ .*

**Prova:** Relembrando o ponto 2 da Propriedade 2.1 podemos ver que

$$|||A^k||| \leq |||A|||^k, \text{ com } k \in \mathbb{N}.$$

Como, por hipótese,  $0 \leq |||A||| < 1$  então

$$\lim_{k \rightarrow +\infty} |||A|||^k = 0$$

donde

$$\lim_{k \rightarrow +\infty} |||A^k||| = 0.$$

Tendo em conta o conhecido resultado, que pode ser visto em [10, p. 298], que diz que sendo  $\{A_k\}_{k \in \mathbb{N}}$  uma sucessão de matrizes, para qualquer norma  $|||\cdot|||$ ,  $\lim_{k \rightarrow \infty} |||A_k||| = 0$  se e só se  $\lim_{k \rightarrow \infty} A_k = 0$ , obtemos o pretendido.  $\square$

**Definição 3.2** *Uma matriz  $A$  diz-se **estável** quando*

$$\lim_{k \rightarrow \infty} A^k = 0.$$

**Teorema 3.2** *Uma matriz  $A \in \mathbb{C}^{n \times n}$  é estável se e só se  $\rho(A) < 1$ .*

**Prova:**

[ $\Rightarrow$ ]

Sejam  $\lim_{k \rightarrow \infty} A^k = 0$  e  $\lambda$  um qualquer valor próprio de  $A$ .

Pelo ponto 3 da Propriedade 2.3,  $A^k x = \lambda^k x$ , com  $x \neq 0$  um vetor próprio associado ao valor próprio  $\lambda$ . Desta forma, como

$$\lim_{k \rightarrow \infty} A^k x = 0,$$

para qualquer  $x \in \mathbb{C}^n \setminus \{0\}$ , tomando  $x$  o vetor próprio associado a  $\lambda$  temos

$$\lim_{k \rightarrow \infty} \lambda^k x = 0.$$

Como  $x \neq 0$  pelo menos uma das suas componentes é não nula, por exemplo  $x_i$ , donde

$$\lim_{k \rightarrow \infty} \lambda^k x_i = 0.$$

Então  $\lim_{k \rightarrow \infty} \lambda^k = 0$  e, portanto,  $|\lambda| < 1$ . Dada a arbitrariedade de  $\lambda$  concluimos que  $\rho(A) < 1$ .

[ $\Leftarrow$ ]

Suponhamos que  $\rho(A) < 1$ . Existe  $\varepsilon > 0$  tal que  $\rho(A) + \varepsilon < 1$  e temos, pelo Lema 3.1,

$$|||A||| \leq \rho(A) + \varepsilon < 1.$$

Usando o ponto 2 da Propriedade 2.1 temos que

$$|||A^k||| \leq |||A|||^k < 1.$$

Pelo Lema 3.2 temos que, para  $k \rightarrow \infty$ ,  $A^k \rightarrow 0$ . □

**Corolário 3.1** *Sendo  $|||\cdot|||$  uma norma matricial, para qualquer matriz  $A \in \mathbb{C}^{n \times n}$*

$$\rho(A) = \lim_{k \rightarrow \infty} |||A^k|||^{\frac{1}{k}}. \quad (3.8)$$

**Prova:** Como

$$\rho(A)^k = \rho(A^k) \leq |||A^k||| \quad (3.9)$$

temos que

$$\rho(A) \leq |||A^k|||^{\frac{1}{k}},$$

para todo o  $k = 1, 2, \dots$

Para  $\varepsilon > 0$ , arbitrário, seja

$$\tilde{A} := [\rho(A) + \varepsilon]^{-1} A.$$

Como

$$\rho(\tilde{A}) = \frac{1}{\rho(A) + \varepsilon} \rho(A) < 1$$

então, pelo Lema 3.2,

$$|||\tilde{A}^k||| \rightarrow 0,$$

quando  $k \rightarrow \infty$ . Assim, existe  $N \in \mathbb{N}$  tal que

$$|||\tilde{A}^k||| < 1, \forall k \geq N. \quad (3.10)$$

Note-se que

$$\begin{aligned} |||A^k||| &= |||((\rho(A) + \varepsilon)\tilde{A})^k||| \\ &= |||(\rho(A) + \varepsilon)^k \tilde{A}^k||| \\ &= (\rho(A) + \varepsilon)^k |||\tilde{A}^k|||. \end{aligned}$$

Devido a (3.10),

$$|||A^k||| \leq [\rho(A) + \varepsilon]^k, \forall k \geq N$$

ou

$$|||A^k|||^{\frac{1}{k}} \leq \rho(A) + \varepsilon, \forall k \geq N. \quad (3.11)$$

Como  $\varepsilon > 0$  é arbitrário e a desigualdade (3.11) se verifica, então  $\lim_{k \rightarrow \infty} |||A^k|||^{\frac{1}{k}}$  existe e é igual a  $\rho(A)$ .  $\square$

O corolário anterior pode ser visto como um “refinamento” da desigualdade (3.1) do Teorema 3.1.

Outros resultados acerca da localização de valores próprios, com recurso a normas, podem ser vistos em [15, pp. 377–379]. Aí, podemos encontrar resultados para a majoração da soma dos valores próprios de uma matriz, assim como para a parte real e/ou para a parte imaginária dos mesmos.

## 3.2 Os Círculos de Gershgorin

As desigualdades (3.5) permitem localizar, no plano complexo, os valores próprios de uma dada matriz numa coroa circular centrada na origem. Outros resultados de localização geométrica podem ser obtidos usando os Círculos de Gershgorin. Estes círculos são fáceis de identificar a partir das entradas da matriz dada.

**Definição 3.3** *Seja  $A = [a_{ij}] \in \mathbb{C}^{n \times n}$  uma matriz dada. Ao círculo, no plano complexo, centrado em  $a_{ii}$  de raio  $r_i(A) = \sum_{j=1, j \neq i}^n |a_{ij}|$  chamamos **círculo de Gershgorin, por linha, centrado em  $a_{ii}$** . Denotamos esse círculo por  $G_i(A)$ , isto é,  $G_i(A) = \{z \in \mathbb{C} : |z - a_{ii}| \leq r_i(A)\}$ .*

**Definição 3.4** *A  $\cup_{i=1}^n G_i(A)$  chamamos **região de Gershgorin de  $A$**  e denotamos por  $G(A)$ .*

**Teorema 3.3 (Círculos de Gershgorin)** *Seja  $A \in \mathbb{C}^{n \times n}$  uma matriz dada. Com as notações das Definições 3.3 e 3.4, as seguintes proposições são verdadeiras:*

1.  $\sigma(A) \subset G(A)$ .

2. Consideremos que  $R_1, \dots, R_n$  é uma sequência de discos de Gershgorin com  $G_1 = \cup_{i=1}^k R_i$  e  $G_2 = \cup_{i=k+1}^n R_i$ , para algum  $k = 1, \dots, n-1$ . Se  $G_1$  e  $G_2$  são tais que  $G_1 \cap G_2 = \emptyset$  e  $G_1$  é conexa, então  $G_1$  contém exatamente  $k$  valores próprios de  $A$ , contando com as suas multiplicidades algébricas, enquanto que os restantes valores próprios estão contidos em  $G_2$ .

**Prova:**

1. A prova deste ponto pode ser vista em [19, p.385] ou em [20, p.184]. Aqui baseamos nas provas de [15, pp. 371–372] e [10, pp. 344–345].

Seja  $\lambda$  um valor próprio de  $A$  e  $x \neq 0$  um vetor próprio associado ao valor próprio  $\lambda$ . Então:

$$\begin{aligned} Ax &= \lambda x \\ \sum_{k=1}^n a_{ik} x_k &= \lambda x_i, \quad i = 1, 2, \dots, n. \end{aligned}$$

Seja  $|x_p| = \max_{i=1, \dots, n} |x_i|$ . Utilizando a  $p$ -ésima equação podemos obter:

$$\begin{aligned} a_{p1}x_1 + \dots + a_{pp}x_p + \dots + a_{pn}x_n &= \lambda x_p \\ \sum_{\substack{k=1 \\ k \neq p}}^n a_{pk}x_k &= \lambda x_p - a_{pp}x_p. \end{aligned}$$

Aplicando módulos, vem que:

$$\begin{aligned} |\lambda - a_{pp}| |x_p| &= \left| \sum_{k=1}^n a_{pk} x_k \right| \\ &\leq \sum_{k=1}^n |a_{pk}| |x_k| \\ &\leq |x_p| \sum_{k=1}^n |a_{pk}|. \end{aligned}$$

Como  $x \neq 0$  então  $|x_p| \neq 0$ , donde

$$|\lambda - a_{pp}| \leq \sum_{k=1, k \neq p}^n |a_{pk}|.$$

2. Vamos considerar a matriz  $A$  escrita da seguinte forma

$$A = D + B$$

em que  $D = \text{diag}(a_{11}, a_{22}, \dots, a_{nn}) = \begin{bmatrix} a_{11} & 0 & \cdots & 0 \\ 0 & a_{22} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & a_{nn} \end{bmatrix}$  e  $B = A - D$  e tomemos

$$A_\varepsilon := D + \varepsilon B, \text{ com } \varepsilon \in [0, 1].$$

Note-se que  $A_0 = D$  e que  $A_1 = A$ . Podemos ver que

$$G(A_\varepsilon) = \cup_{i=1}^n \{z \in \mathbb{C} : |z - a_{ii}| \leq r_i(A_\varepsilon)\}$$

e como  $r_i(A_\varepsilon) = \varepsilon r_i(A)$ , temos que

$$G(A_\varepsilon) = \cup_{i=1}^n \{z \in \mathbb{C} : |z - a_{ii}| \leq \varepsilon r_i(A)\}.$$

Sem perda de generalidade, suponhamos que os  $k$  discos que formam  $G_1$  são os primeiros  $k$  discos, ou seja,

$$G_1 = \cup_{i=1}^k \{z \in \mathbb{C} : |z - a_{ii}| \leq r_i(A)\},$$

e que  $G_1$  é uma região conexa. Além disso,

$$G_2 = \cup_{i=k+1}^n \{z \in \mathbb{C} : |z - a_{ii}| \leq r_i(A)\}$$

é formada pelos restantes discos  $k+1, \dots, n$ .

Note-se que

$$G_1(A_\varepsilon) = \cup_{i=1}^k \{z \in \mathbb{C} : |z - a_{ii}| \leq \varepsilon r_i(A)\}$$

é tal que

$$G_1(A_\varepsilon) \subset G_1, \text{ para todo o } \varepsilon \in [0, 1].$$

Da mesma forma podemos ver que

$$G_2(A_\varepsilon) \subset G_2, \text{ para todo o } \varepsilon \in [0, 1],$$

e portanto

$$G_2(A_\varepsilon) \cap G_1 = \emptyset, \forall \varepsilon \in [0, 1].$$

Para cada  $i = 1, \dots, k$  seja  $\lambda_i(A_0) = a_{ii}$ . Os valores próprios de uma matriz são funções contínuas das suas entradas, ver [22, secção 3.1.1]. Assim, a função  $\lambda_i(\varepsilon) = \lambda_i(A_\varepsilon)$  é contínua e é tal que  $\lambda_i(0) = \lambda_i(A_0) = a_{ii}$  e  $\lambda_i(1) = \lambda_i(A_1) = \lambda_i(A)$ . Como, para cada  $\varepsilon \in [0, 1]$ ,

$$\lambda_i(\varepsilon) \in G_1(A_\varepsilon)$$

e  $G_1(A_\varepsilon) \subset G_1$ , então  $\lambda_i(\varepsilon)$  forma uma linha contínua em  $G_1$  que liga  $\lambda_i(A_0)$  a  $\lambda_i(A)$ . Deste modo, garantimos que existem  $k$  valores próprios de  $A$  em  $G_1$ . Não podem existir mais pois, tomando qualquer outro dos valores próprios de  $A_0$ , ele pertence a um dos discos de  $G_2$ . Como  $G_2(A_\varepsilon) \subset G_2$ , para qualquer  $\varepsilon \in [0, 1]$ , os correspondentes valores próprios  $\lambda_i(A_\varepsilon)$  formam uma curva contínua em  $G_2$ , que é disjunta de  $G_1$ . Assim, o valor próprio correspondente de  $A$  não poderá pertencer a  $G_1$ .

□

Relembrando que os valores próprios de  $A$  são os valores próprios de  $A^*$ , do Teorema 3.3 pode ser obtido um resultado análogo envolvendo as colunas de  $A$ . Isto é, podemos concluir que:

$$\sigma(A) \subset \cup_{j=1}^n C_j = G'(A)$$

em que  $C_j$  é um círculo, do plano complexo, com centro em  $|a_{jj}|$  e raio  $r_j(A) = \sum_{i=1, i \neq j}^n |a_{ij}|$  e  $G'(A)$  denota os círculos de Gershgorin, por coluna.

**Corolário 3.2** Para uma dada matriz  $A \in C^{n \times n}$ ,

$$\forall \lambda \in \sigma(A), \quad \lambda \in G(A) \cap G'(A).$$

### 3.3 Exemplos

**Exemplo 3.1** Tomemos a matriz simétrica

$$\begin{bmatrix} 1 & 2 & 5 & 1 \\ 2 & 3 & 4 & 3 \\ 5 & 4 & 5 & 1 \\ 1 & 3 & 1 & 4 \end{bmatrix}. \quad (3.12)$$

Começemos por aplicar os resultados enunciados com recurso a normas.

Apliquemos o Teorema 3.1. Usando a equação (3.1) e considerando a norma induzida  $\|\cdot\|_1$ , temos que:

$$\|A\|_1 = \max_{1 \leq j \leq 4} \left( \sum_{i=1}^4 |a_{ij}| \right) = \max\{9, 12, 15, 9\} = 15$$

donde

$$\rho(A) \leq 15.$$

Com o auxílio do MATLAB® conseguimos determinar a matriz inversa de  $A$ ,

$$A^{-1} = \begin{bmatrix} -\frac{7}{9} & \frac{19}{18} & \frac{1}{18} & -\frac{11}{18} \\ \frac{19}{18} & -\frac{19}{9} & \frac{7}{18} & \frac{11}{9} \\ \frac{1}{18} & \frac{7}{18} & -\frac{1}{9} & -\frac{5}{18} \\ -\frac{11}{18} & \frac{11}{9} & -\frac{5}{18} & -\frac{4}{9} \end{bmatrix}$$

Utilizando o comando `norm(M,1)`, do MATLAB®, que calcula  $\|M\|_1$ , temos que  $\|A^{-1}\|_1 \cong 4.777777777777773$ , donde, aplicando a Proposição 3.1, vem que

$$\ell(A) = \frac{1}{\|A^{-1}\|_1} \cong 0.209302325581396.$$



Daqui podemos dizer que para qualquer valor próprio,  $\lambda$ , da matriz  $A$

$$0.209302325581396 \leq |\lambda| \leq 15.$$

Agora vamos determinar os círculos de Gershgorin relativos a esta matriz. Os centros e raios de  $A$  e  $A^T$  são os mesmos, visto que a matriz é simétrica, e temos:

<b>Centros</b>	<b>Raios</b>
1	8
3	9
5	10
4	5

Pela Propriedade 2.4 sabemos que esta matriz possui apenas valores próprios reais e, ao observar os círculos na Figura 3.1, verificamos que os valores próprios reais de  $A$  estão contidos no intervalo  $[-7, 15]$ .

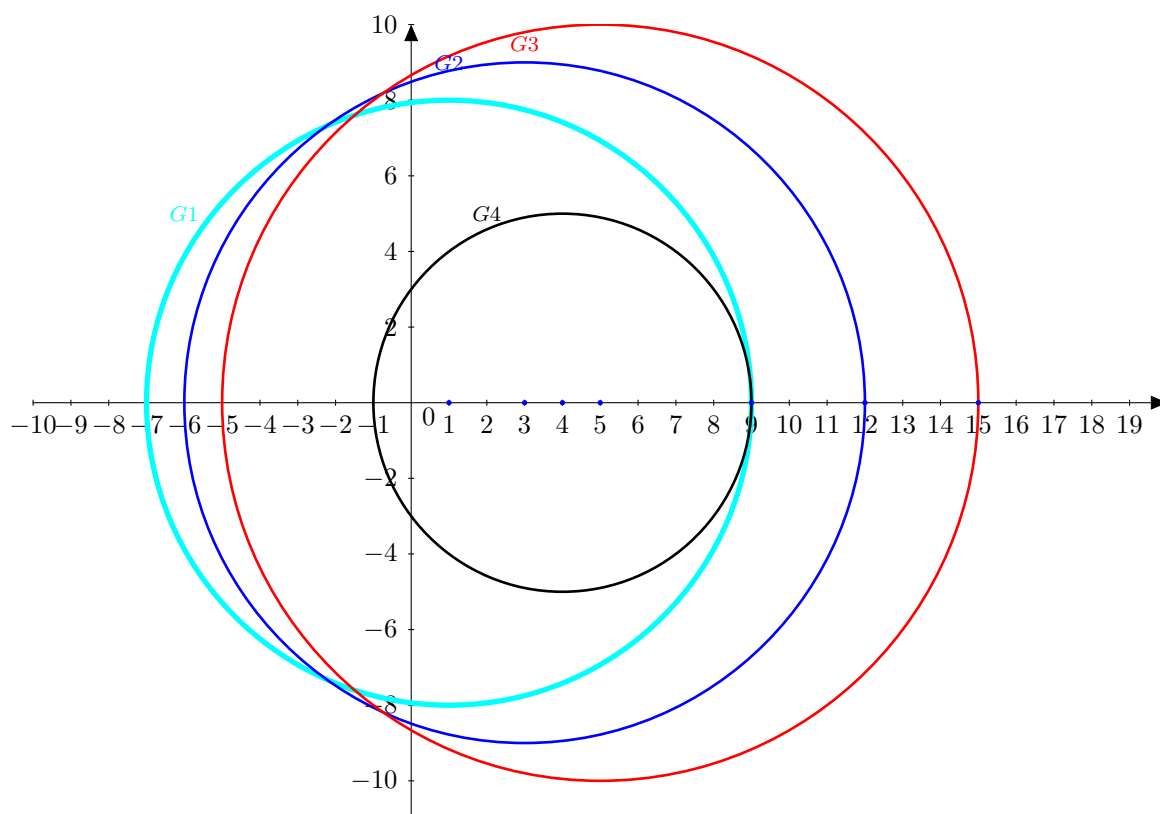


Figura 3.1: Círculos de Gershgorin – matriz  $A$

Podemos confirmar os valores obtidos usando o comando `eig(M)`, do MATLAB®, que devolve uma aproximação dos valores próprios da matriz  $M$ . Assim, uma aproximação obtida para os valores próprios da matriz  $A$  é a seguinte:

$$\begin{aligned}\lambda_1 &\cong -2.563382668195001 \\ \lambda_2 &\cong -0.295188571810782 \\ \lambda_2 &\cong 4.018097046416818 \\ \lambda_2 &\cong 11.840474193588962.\end{aligned}\tag{3.13}$$

**Exemplo 3.2** *Tomemos a matriz*

$$B = \begin{bmatrix} 30 & 1 & 0 & 2 \\ 1 & 10 & 2 & 1 \\ 0 & 1 & 4 & 0 \\ 4 & 0 & -5 & 9 \end{bmatrix}.$$

*Temos que*

$$\|B\|_\infty = 33$$

*donde*

$$\rho(B) \leq 33.$$

*Usando a equação (3.2) obtemos uma majoração mais grosseira, isto é, conclui-se que*

$$\begin{aligned}\rho(B)^2 &\leq 1155 \\ \rho(B) &\cong 33.985290935932859.\end{aligned}$$

*Temos, ainda, que*

$$\ell(B) = \frac{1}{\|B^{-1}\|_\infty} \cong 3.366323024054982.$$

*Donde concluímos que, para qualquer valor próprio,  $\lambda$ , da matriz  $B$*

$$3.366323024054982 \leq |\lambda| \leq 33.$$

*Vamos agora determinar os Círculos de Gershgorin relativos a esta matriz. Os centros e raios para a matriz  $B$  são os seguintes*

<i><b>Centros</b></i>	<i><b>Raios</b></i>
30	3
10	4
4	1
9	9

e obtemos os círculos representados na Figura 3.2. Como podemos observar a região de Gershgorin da matriz  $B$  é a união das regiões  $G_{1,B} = G_2(B) \cup G_3(B) \cup G_4(B)$  e  $G_{2,B} = G_1(B)$ , disjuntas. Assim, existem três valores próprios no círculo centrado em  $(9, 0)$  e raio 9 e um valor próprio no círculo de centro em  $(3, 0)$  e raio 3. Deste modo, também podemos concluir que  $B$  tem um valor próprio real no intervalo  $[27, 30]$  e um dos valores próprios restantes é, necessariamente, real e pertence ao intervalo  $[0, 18]$ .

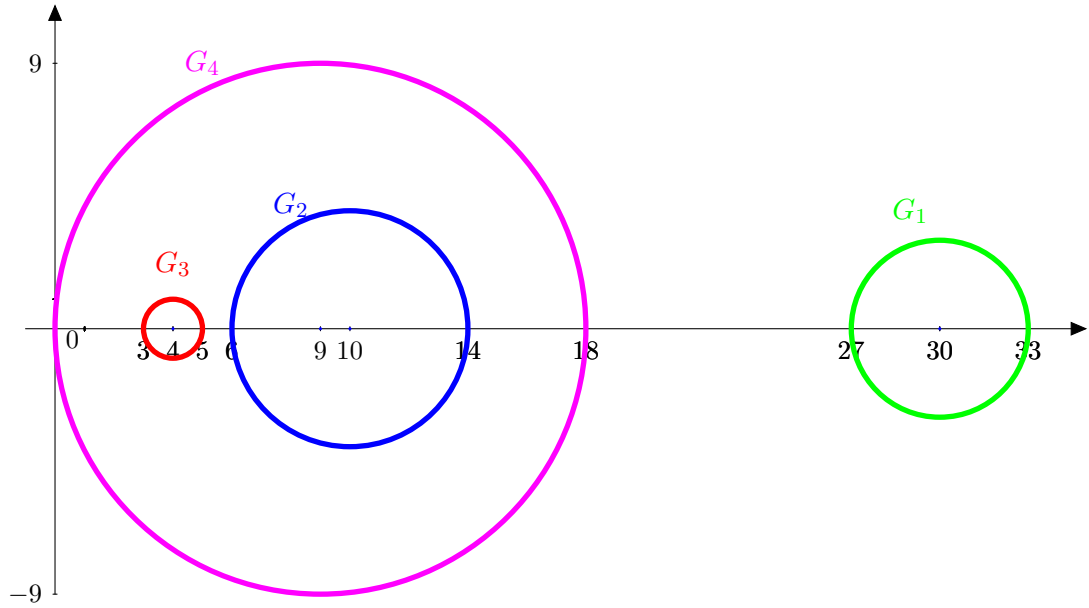


Figura 3.2: Círculos de Gershgorin – matriz  $B$

Usando o MATLAB<sup>®</sup> encontramos aproximações para os valores próprios da matriz  $B$ :

$$\begin{aligned}
 \lambda_1 &\cong 30.430542981441548 \\
 \lambda_2 &\cong 9.509741443548016 + 0.495291391851072 i \\
 \lambda_2 &\cong 9.509741443548016 - 0.495291391851072 i \\
 \lambda_2 &\cong 3.549974131462413,
 \end{aligned} \tag{3.14}$$

que confirmam a análise feita.



## Capítulo 4

# Obtenção do polinómio caraterístico, dada uma matriz

Como vimos atrás, o polinómio caraterístico de uma dada matriz  $A \in \mathbb{C}^{n \times n}$  é dado por:

$$p_A(\lambda) = \det(\lambda I_n - A) = \lambda^n + p_{n-1}\lambda^{n-1} + \cdots + p_0. \quad (4.1)$$

Em [11, Cap. 6], o autor refere métodos diretos de determinação exata do polinómio caraterístico e métodos iterativos que nos levam a uma aproximação deste. Neste último caso, e devido ao erro de truncatura, os resultados não estarão só afetados de erros de arredondamento.

Neste capítulo, abordaremos quatro processos algorítmicos, clássicos, de obtenção do polinómio caraterístico de uma matriz, visto que o cálculo direto do determinante, por exemplo pelo Teorema de Laplace, torna-se complicado quando  $n$  é muito grande, pois o número de operações aumenta exponencialmente.

Os métodos aqui focados serão o Método de Danilevsky, o Método de Krylov, o Método de Leverrier e o Método dos Coeficientes Indeterminados que podem ser vistos, por exemplo, em [11, Cap.6], [4, Cap.6] ou em [27].

### 4.1 O método de Danilevsky

O objetivo deste método é o de transformar uma matriz  $A$ , dada, numa matriz semelhante triangular por blocos, em que os blocos da diagonal são matrizes companheiras. Este método obtém, para algumas matrizes, e caso isso seja possível, a sua matriz companheira. Iremos, em primeiro lugar, desenvolver o método para este caso. Mais à frente veremos como proceder no caso em que não é possível obter a matriz companheira de  $A$  e o método produz o aparecimento da tal matriz triangular por blocos.

**1º Caso** O método produz a matriz companheira.

Pretende-se transformar por semelhança a matriz  $A$  na forma

$$C_A = \begin{bmatrix} -p_{n-1} & -p_{n-2} & \cdots & -p_1 & -p_0 \\ 1 & 0 & \cdots & 0 & 0 \\ 0 & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 1 & 0 \end{bmatrix}, \quad (4.2)$$

em que diremos que as linhas  $n \geq 2$  estão reduzidas.

Desta forma,

$$\det(\lambda I_n - C_A) = \lambda^n + p_{n-1}\lambda^{n-1} + p_{n-2}\lambda^{n-2} + \cdots + p_1\lambda + p_0. \quad (4.3)$$

Como a transformação é de semelhança, este polinómio é o polinómio caraterístico de  $A$ .

O método de Danilevski consiste em efetuar  $(n - 1)$  transformações de semelhança, a partir da matriz  $A$ , encontrando matrizes  $M_i, i = 1, \dots, n - 1$ , não singulares, tais que  $C_A = M_1^{-1} \dots M_{n-1}^{-1} A M_{n-1} \dots M_1$ , ou seja, a matriz  $A$  vai sendo, sucessivamente, substituída por matrizes cujas linhas vão sendo reduzidas, a partir da linha  $n$  até à linha 2. Este procedimento produzirá a matriz  $C_A$  se determinados elementos das matrizes, entretanto obtidas, forem não nulos. A esses elementos chamaremos, como usualmente, pivots.

Os passos seguintes correspondem à redução, sequencial, das linhas  $(n - 1)$  à 2 à forma  $[0 \ \cdots \ 0 \ 1 \ 0 \ 0]$  a  $[1 \ 0 \ \cdots \ 0]$ , respetivamente. Este processo de redução das linhas será possível desde que, para cada linha, se possa encontrar um pivot.

Vamos começar por descrever o primeiro passo da redução que corresponde à redução da linha  $n$  da matriz  $A$  à forma  $[0 \ \cdots \ 0 \ 1 \ 0]$ .

Seja a matriz:

$$A = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{bmatrix}.$$

o processo tem início na construção da matriz  $M_{n-1}$ . Caso  $a_{n,n-1} = 0$ , suponha-se que existe  $a_{nk} \neq 0$ , com  $1 \leq k \leq n - 2$ , então troca-se a coluna  $k$  pela coluna  $(n - 1)$  e a linha  $k$  pela  $(n - 1)$ , obtendo-se uma matriz  $\tilde{A} = [\tilde{a}_{ij}]$  semelhante a  $A$  onde  $\tilde{a}_{n,n-1} \neq 0$ . O caso em que na linha  $(n - 1)$  não existem elementos não nulos, da coluna 1 à  $(n - 1)$ , cai no segundo caso, que trataremos adiante.

Depois, executa-se o procedimento para  $A$  sob a hipótese de  $a_{n,n-1} \neq 0$ , tal como se descreve de seguida.

Toma-se:

$$M_{n-1} = \left[ \begin{array}{cccc|c} 1 & \cdots & 0 & 0 & 0 \\ \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & \cdots & 1 & 0 & 0 \\ \hline m_{n-1,1} & \cdots & m_{n-1,n-2} & m_{n-1,n-1} & m_{n-1,n} \\ 0 & \cdots & 0 & 0 & 1 \end{array} \right] \quad (4.4)$$

onde

$$m_{n-1,i} = -\frac{a_{ni}}{a_{n,n-1}} \quad \text{para } i \neq n-1 \quad (4.5)$$

e

$$m_{n-1,n-1} = \frac{1}{a_{n,n-1}}, \quad (4.6)$$

ou seja, a  $(n-1)$ -ésima linha de  $M_{n-1}$  é:

$$-\frac{a_{n1}}{a_{n,n-1}}, \quad -\frac{a_{n2}}{a_{n,n-1}}, \quad \dots, \quad \frac{1}{a_{n,n-1}}, \quad -\frac{a_{nn}}{a_{n,n-1}}.$$

Ao multiplicar-se  $A$  por  $M_{n-1}$  obtemos uma matriz  $B_{n-1} \in \mathbb{C}^{n \times n}$ . Aplicando as regras da multiplicação de matrizes podemos ver que a última linha de  $B_{n-1}$  é igual a

$$0 \quad 0 \quad \cdots \quad 1 \quad 0.$$

A matriz  $M_{n-1}$  é invertível. De facto, ela pode ser vista como produto de matrizes elementares. Mais, a matriz  $M_{n-1}^{-1}$  é:

$$M_{n-1}^{-1} = \left[ \begin{array}{ccccc} 1 & 0 & \cdots & 0 & 0 \\ 0 & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ -\frac{1}{m_{n-1,1}} & -\frac{1}{m_{n-1,2}} & \cdots & \frac{1}{m_{n-1,n-1}} & -\frac{1}{m_{n-1,n}} \\ 0 & 0 & \cdots & 0 & 1 \end{array} \right]. \quad (4.7)$$

Agora podemos obter a matriz semelhante a  $A$ ,  $D_{n-1} = M_{n-1}^{-1} A M_{n-1} = M_{n-1}^{-1} B_{n-1}$ . Observe-se que as linhas das matrizes  $M_{n-1}$  e  $M_{n-1}^{-1}$  são compostas por vetores unitários, exceto na  $(n-1)$ -ésima linha. Tendo isto em conta, é possível mostrar

que  $D_{n-1}$  tem a seguinte forma:

$$D_{n-1} = \begin{bmatrix} d_{1,1}^{(n-1)} & d_{1,2}^{(n-1)} & \cdots & d_{1,n-1}^{(n-1)} & d_{1,n}^{(n-1)} \\ d_{2,1}^{(n-1)} & d_{2,2}^{(n-1)} & \cdots & d_{2,n-1}^{(n-1)} & d_{2,n}^{(n-1)} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ d_{n-1,1}^{(n-1)} & d_{n-1,2}^{(n-1)} & \cdots & d_{n-1,n-1}^{(n-1)} & d_{n-1,n}^{(n-1)} \\ 0 & 0 & \cdots & 1 & 0 \end{bmatrix}. \quad (4.8)$$

De facto, a multiplicação de  $M_{n-1}^{-1}$  por  $B_{n-1}$  apenas altera a  $(n-1)$ -ésima linha de  $B_{n-1}$  e, portanto, a matriz  $D_{n-1}$  tem a mesma forma que  $B_{n-1}$ , isto é, a matriz  $D_{n-1}$  é semelhante a  $A$  e tem uma linha reduzida.

Se na linha  $(n-1)$  existir algum elemento não nulo, da coluna 1 até  $(n-2)$ , escolhemos um desses elementos para pivot e continuamos tal como descrito para a linha  $n$ , com as devidas adaptações, e assim sucessivamente até à linha 2. Mais precisamente, consideremos a matriz reduzida de  $A$ , de baixo para cima, até à linha  $(n-k)$ , para algum  $1 \leq k \leq n-2$ , como sendo  $D_{n-k}$ . O processo de redução da linha  $(n-k-1)$ , neste primeiro caso, é feito da seguinte forma:

- escolhemos  $d_{n-k-1,j}^{(n-k)} \neq 0$ , para algum  $1 \leq j \leq n-k-2$ . Este passa a ser o pivot desta linha. Esta escolha de pivot corresponde à troca, na matriz  $D_{n-k}$ , da coluna  $j$  pela  $(n-k-2)$  e da linha  $j$  pela  $(n-k-2)$ . Com esta troca de linhas e colunas obtemos uma matriz semelhante a  $D_{n-k}$  em que as linhas reduzidas não foram alteradas. Denotemos esta matriz por  $\tilde{D}_{n-k}$ . Observe-se que  $D_{n-k}$  pode ser, eventualmente, igual a  $\tilde{D}_{n-k}$  por não haver necessidade de trocas de pivot;
- a matriz reduzida, obtida nesta iteração do método, é

$$D_{n-k-1} = M_{n-k-1}^{-1} \tilde{D}_{n-k} M_{n-k-1}$$

onde

$$M_{n-k-1} = \left[ \begin{array}{cccc|ccc} 1 & \cdots & 0 & 0 & 0 & \cdots & 0 \\ 0 & \cdots & 0 & 0 & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & 1 & 0 & 0 & \cdots & 0 \\ \hline m_{n-k-1,1} & \cdots & m_{n-k-1,n-k-2} & m_{n-k-1,n-k-1} & m_{n-k-1,n-k} & \cdots & m_{n-k-1,n} \\ 0 & \cdots & 0 & 0 & 1 & \cdots & 0 \\ \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & 0 & 0 & \cdots & 1 \end{array} \right]$$

em que

$$m_{n-k-1,i} = -\frac{\tilde{d}_{n-k,i}^{(n-k)}}{\tilde{d}_{n-k,n-k-1}^{(n-k)}} \quad \text{para } i \neq n-k-1 \quad (4.9)$$



e

$$m_{n-k-1,n-k-1} = \frac{1}{\tilde{d}_{n-k,n-k-1}^{(n-k)}}. \quad (4.10)$$

Tal como para a linha  $n$ , este procedimento produz uma matriz semelhante, mantendo as linhas previamente reduzidas.

**2º Caso** O método produz uma matriz triangular por blocos companheiros.

Este caso ocorre, quando em alguma das iterações do método descrito no primeiro caso, a linha a reduzir não tem pivots. Assim, suponhamos que essa situação ocorre na redução da linha  $(n-k)$ , ou seja, a matriz  $D_{n-k}$  é tal que  $d_{n-k,j}^{(n-k)} = 0$ ,  $1 \leq j \leq n-k-1$ , isto é,

$$D_{n-k} = \left[ \begin{array}{ccc|ccc} d_{11}^{(n-k)} & \cdots & d_{1,n-k-2}^{(n-k)} & d_{1,n-k-1}^{(n-k)} & \cdots & d_{1n}^{(n-k)} \\ \vdots & \ddots & \ddots & \vdots & \ddots & \vdots \\ d_{n-k-1,1}^{(n-k)} & \cdots & d_{n-k-1,n-k-1}^{(n-k)} & d_{n-k-1,n-k}^{(n-k)} & \cdots & d_{n-k-1,n}^{(n-k)} \\ \hline 0 & \cdots & 0 & d_{n-k,n-k}^{(n-k)} & \cdots & d_{n-k,n}^{(n-k)} \\ 0 & \cdots & 0 & 1 & \cdots & 0 \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & 0 & \cdots & 1 \end{array} \right].$$

Note-se que

$$D_{n-k} = \left[ \begin{array}{c|c} D & * \\ \hline 0 & C \end{array} \right],$$

onde  $C$  é uma matriz companheira.

A redução prossegue usando na matriz  $D$  o processo descrito para o primeiro caso, ou para o segundo caso, até se obter uma matriz da forma

$$\left[ \begin{array}{cccc} C_p & * & \cdots & * \\ \vdots & \ddots & & \vdots \\ 0 & \cdots & C_2 & * \\ 0 & 0 & \cdots & C_1 \end{array} \right]$$

onde  $C_1, C_2, \dots, C_p$ , com  $2 \leq p \leq n$ , são matrizes companheiras. Neste caso, o  $\det(\lambda I_n - A)$  fatoriza-se em  $p$  polinómios caraterísticos correspondentes às matrizes  $C_1, C_2, \dots, C_p$ , isto é:

$$\det(\lambda I_n - A) = \det(\lambda I - C_1) \cdots \det(\lambda I - C_p).$$

Para uma melhor compreensão do método de Danilevsky, apresentamos dois exemplos.

**Exemplo 4.1** *Seja*

$$A = \begin{bmatrix} 1 & 2 & 5 & 1 \\ 2 & 3 & 4 & 3 \\ 5 & 4 & 5 & 1 \\ 1 & 3 & 1 & 4 \end{bmatrix}. \quad (4.11)$$

Com o auxílio do MATLAB<sup>®</sup>, obtemos o seguinte:

- *Primeira iteração: pivot igual a 1*

$$M_3 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ -1 & -3 & 1 & -4 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

A matriz  $D_3$  semelhante a  $A$ , com uma linha reduzida é:

$$D_3 = \begin{bmatrix} -4 & -13 & 5 & -19 \\ -2 & -9 & 4 & -13 \\ -10 & -51 & 26 & -77 \\ 0 & 0 & 1 & 0 \end{bmatrix}.$$

- *Segunda iteração: pivot igual a  $-51$*

$$M_2 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ -\frac{10}{51} & -\frac{1}{51} & \frac{26}{51} & -\frac{77}{51} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

A matriz  $D_2$  semelhante a  $A$ , com duas linha reduzidas é:

$$D_2 = \begin{bmatrix} -\frac{74}{51} & \frac{13}{51} & -\frac{83}{51} & \frac{32}{51} \\ \frac{1352}{51} & \frac{737}{51} & -\frac{1567}{51} & -\frac{1850}{51} \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}.$$

- Terceira iteração: pivot igual a  $\frac{1352}{51}$

$$M_1 = \begin{bmatrix} \frac{51}{1352} & -\frac{737}{1352} & \frac{1567}{1352} & \frac{925}{676} \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

A matriz  $D_1$  semelhante a  $A$ , com três linha reduzidas é:

$$D_1 = \begin{bmatrix} 13 & -3 & -124 & -36 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

Assim, conclui-se que o polinómio caraterístico da matriz  $A$  é:

$$p(x) = x^4 - 13x^3 + 3x^2 + 124x + 36. \quad (4.12)$$

**Exemplo 4.2** Tomemos agora a matriz

$$B = \begin{bmatrix} 10 & 9 & 12 & 3 \\ -8 & -8 & -11 & -2 \\ 2 & 3 & 4 & 0 \\ -11 & -12 & -15 & -2 \end{bmatrix}.$$

Aplicando o método de Danilevsky, obtemos, após a primeira iteração a matriz:

$$D_3 = \left[ \begin{array}{cc|cc} \frac{6}{5} & -\frac{3}{5} & -\frac{4}{5} & \frac{7}{5} \\ \frac{1}{15} & \frac{4}{5} & \frac{11}{15} & -\frac{8}{15} \\ 0 & 0 & 2 & -1 \\ 0 & 0 & 1 & 0 \end{array} \right].$$

Para a segunda iteração entramos no segundo caso, explicitado acima, onde vamos tomar:

$$C_1 = \begin{bmatrix} 2 & -1 \\ 1 & 0 \end{bmatrix}$$

e

$$D = \left[ \begin{array}{cc} \frac{6}{5} & -\frac{3}{5} \\ \frac{1}{15} & \frac{4}{5} \end{array} \right].$$

Desta forma, podemos continuar a aplicar o método para a matriz  $D$ , obtendo-se:

$$C_2 = \begin{bmatrix} 2 & -1 \\ 1 & 0 \end{bmatrix}.$$

e, assim, o polinómio caraterístico da matriz  $B$  é dado por:

$$\begin{aligned} p_B(\lambda) &= \det(\lambda I_2 - C_1) \det(\lambda I_2 - C_2) \\ &= (\lambda^2 - 2\lambda + 1)(\lambda^2 - 2\lambda + 1). \end{aligned}$$

## 4.2 O método de Krylov

Retomemos o polinómio caraterístico de uma dada matriz  $A$ , (4.1). Pelo teorema de Cayley-Hamilton temos que  $p_A(A) = 0$ , ou seja,

$$A^n + p_{n-1}A^{n-1} + \cdots + p_0I_n = 0. \quad (4.13)$$

O método de Krylov inicia-se tomando para vetor inicial um qualquer vetor,  $v_0$ , não nulo, em que  $v_0 \in \mathbb{C}^n$ . Se na equação (4.13) multiplicarmos ambos os membros por  $v_0$  vem que:

$$A^n v_0 + p_{n-1}A^{n-1}v_0 + \cdots + p_0v_0 = 0. \quad (4.14)$$

Façamos

$$A^k v_0 = W_k, \text{ com } k = 0, 1, \dots, n. \quad (4.15)$$

Substituindo na equação (4.14) vem que:

$$W_n + p_{n-1}W_{n-1} + \cdots + p_0W_0 = 0,$$

ou seja,

$$p_{n-1}W_{n-1} + \cdots + p_0W_0 = -W_n.$$

A fórmula (4.15) pode ser escrita na forma recursiva, do seguinte modo:

$$\begin{cases} W_k &= AW_{k-1} & \text{com } k = 1, \dots, n \\ W_0 &= v_0 \end{cases}.$$

Matricialmente, temos

$$\begin{bmatrix} W_{n-1} & W_{n-2} & \cdots & W_0 \end{bmatrix} \begin{bmatrix} p_{n-1} \\ p_{n-2} \\ \vdots \\ p_0 \end{bmatrix} = -W_n. \quad (4.16)$$

Tomando  $W = \begin{bmatrix} W_{n-1} & W_{n-2} & \cdots & W_0 \end{bmatrix}$  e  $b = -W_n$  os coeficientes do polinómio caraterístico de  $A$  obtêm-se resolvendo o sistema de equações lineares

$$WP = b. \quad (4.17)$$

Assim, se  $W$  for invertível, o sistema (4.17) é possível e determinado e

$$P = W^{-1}b. \quad (4.18)$$

Caso contrário, se  $W$  não for invertível, o sistema (4.17) não tem solução única e não é possível identificar os coeficientes do polinómio caraterístico. Neste caso, escolhe-se outro vetor inicial  $v_0 \in \mathbb{C}$ , não nulo, e repete-se o procedimento de modo a que este sistema venha a ser possível e determinado.

**Exemplo 4.3** *Para uma melhor compreensão do método de Krylov, apresentamos agora um exemplo.*

*Vamos tomar, novamente, a matriz (4.11), como vetor inicial  $v_0 = [1 \ 0 \ 0 \ 0]^T$  e, com o auxílio do MATLAB® obtemos o seguinte*

$$W_1 = Av_0 = \begin{bmatrix} 1 \\ 2 \\ 5 \\ 1 \end{bmatrix},$$

$$W_2 = AW_1 = \begin{bmatrix} 31 \\ 31 \\ 39 \\ 16 \end{bmatrix},$$

$$W_3 = AW_2 = \begin{bmatrix} 304 \\ 359 \\ 490 \\ 227 \end{bmatrix}$$

e

$$W_4 = AW_3 = \begin{bmatrix} 3699 \\ 4326 \\ 5633 \\ 2779 \end{bmatrix}.$$

*Criamos a matriz  $W$  formada pelos vetores  $W_k$ ,  $k = 1, 2, 3$ , determinados e pelo vetor inicial*

$$W = [W_3 \ W_2 \ W_1 \ v_0]$$

*e tomamos o vetor  $b = -W_4$ .*

*Aplicando o método temos, para os coeficientes do polinómio caraterístico, o seguinte:*

$$\begin{aligned} P &= W^{-1}b \\ &= \begin{bmatrix} -13 \\ 3 \\ 124 \\ 36 \end{bmatrix}. \end{aligned}$$

*Assim, o polinómio caraterístico da matriz  $A$  é:*

$$p(x) = x^4 - 13x^3 + 3x^2 + 124x + 36.$$

### 4.3 O método de Leverrier

Consideremos a matriz  $A = [a_{ij}]$  e seja (4.1) o seu polinómio caraterístico. Sejam  $\lambda_1, \dots, \lambda_n$  todas as raízes do polinómio, contando com as suas multiplicidades. O método de Leverrier baseia-se no cálculo do traço das matrizes  $A^k$ ,  $k = 1, \dots, n$ , e nas fórmulas de Newton ([7, p. 241] ou [14, pp.322–324]) para a identificação dos coeficientes de um polinómio, dadas as suas raízes. Isto é, se  $\lambda_1, \dots, \lambda_n$  forem as raízes, contadas as multiplicidades, de um polinómio mónico de grau  $n$ ,  $p(\lambda)$ , as fórmulas de Newton são as seguintes

$$s_k + p_{n-1}s_{k-1} + \dots + p_{n-(k-1)}s_1 = -kp_{n-k}$$

onde

$$s_k = \lambda_1^k + \lambda_2^k + \dots + \lambda_n^k \text{ com } k = 0, \dots, n,$$

e

$$\begin{aligned} p_{n-1} &= -s_1 \\ p_{n-2} &= -\frac{1}{2}(s_2 + p_{n-1}s_1) \\ &\vdots \\ p_0 &= -\frac{1}{n}(s_n + p_{n-1}s_{n-1} + \dots + p_1s_1). \end{aligned} \tag{4.19}$$

É fácil de constatar que, conhecidos os  $s_k$ , os coeficientes do polinómio podem ser calculados, recursivamente, fazendo  $k = 1, \dots, n$ . Ora, pelo Teorema 2.1 e pelo ponto 3 da Propriedade 2.3, temos

$$\begin{aligned} \text{tr}(A^k) &= \sum_{i=1}^n \lambda_i^k \\ &= s_k. \end{aligned}$$

Uma das vantagens do método de Leverrier é que o procedimento é muito simples e não existem casos excecionais. No entanto, o método é trabalhoso visto que é necessário calcular elevadas potências da matriz.

**Exemplo 4.4** *Para uma melhor compreensão do método de Leverrier, apresentamos agora um exemplo.*

*Vamos tomar a matriz (4.11) e, com o auxílio do MATLAB®, obtemos:*

- *potências da matriz A:*

$$A^2 = \begin{bmatrix} 31 & 31 & 39 & 16 \\ 31 & 38 & 45 & 27 \\ 39 & 45 & 67 & 26 \\ 16 & 27 & 26 & 27 \end{bmatrix},$$

$$A^3 = \begin{bmatrix} 304 & 359 & 490 & 227 \\ 359 & 437 & 559 & 298 \\ 490 & 559 & 736 & 345 \\ 227 & 298 & 345 & 231 \end{bmatrix}$$

e

$$A^4 = \begin{bmatrix} 3699 & 4326 & 5633 & 2779 \\ 4326 & 5159 & 6636 & 3421 \\ 5633 & 6636 & 8711 & 4283 \\ 2779 & 3421 & 4283 & 2390 \end{bmatrix};$$

- cálculo do traço das matrizes  $A^k = s_k$ ,  $k = 1, 2, 3, 4$ ,

$$s_1 = 13,$$

$$s_2 = 163,$$

$$s_3 = 1708$$

e

$$s_4 = 19959;$$

- cálculo dos coeficientes do polinómio caraterístico, usando as fórmulas (4.19),

$$p_3 = -13,$$

$$p_2 = 3,$$

$$p_1 = 124$$

e

$$p_0 = 36.$$

Aplicado o método de Leverrier, o polinómio caraterístico da matriz  $A$  é:

$$p(x) = x^4 - 13x^3 + 3x^2 + 124x + 36.$$

## 4.4 O método dos coeficientes indeterminados

Tomemos, de novo, em atenção o polinómio caraterístico de uma matriz  $A$  dado por (4.1). O método dos coeficientes indeterminados começa por substituir, sucessivamente,  $\lambda$  por  $0, 1, \dots, n-1$ , ou seja,

$$\left\{ \begin{array}{rcl} p_0 & = & p_A(0) \\ 1^n + p_{n-1} + \dots + p_1 + p(0) & = & p_A(1) \\ 2^n + 2^{n-1}p_{n-1} + \dots + 2p_1 + p(0) & = & p_A(2) \\ & \vdots & \\ (n-1)^n + (n-1)^{n-1}p_{n-1} + \dots + (n-1)p_1 + p(0) & = & p_A(n-1) \end{array} \right. \quad (4.20)$$

que podemos escrever como

$$\left\{ \begin{array}{rcl} p_0 & = & p_A(0) \\ p_{n-1} + \cdots + p_1 + p(0) & = & p_A(1) - 1^n \\ 2^{n-1}p_{n-1} + \cdots + 2p_1 + p(0) & = & p_A(2) - 2^n \\ & \vdots & \\ (n-1)^{n-1}p_{n-1} + \cdots + (n-1)p_1 + p(0) & = & p_A(n-1) - (n-1)^n \end{array} \right., \quad (4.21)$$

onde  $p_A(k) = \det(kI_n - A)$ ,  $0 \leq k \leq n-1$ .

Desta forma, encontramos, para os coeficientes  $p_0, p_1, \dots, p_{n-1}$ , um sistema de equações possível de ser resolvido. Matricialmente, temos:

$$\begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ 1 & 1 & 1 & \cdots & 1 \\ 1 & 2 & 2^{n-2} & \cdots & 2^{n-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & n-1 & (n-1)^2 & \cdots & (n-1)^{n-1} \end{bmatrix} \begin{bmatrix} p_0 \\ p_1 \\ p_2 \\ \vdots \\ p_{n-1} \end{bmatrix} = \begin{bmatrix} p_A(0) \\ p_A(1) - 1^n \\ p_A(2) - 2^n \\ \vdots \\ p_A(n-1) - (n-1)^n \end{bmatrix}$$

ou

$$VP = d.$$

Como a matriz  $V$  é uma matriz de Vandermonde, é invertível (ver em [7, pp. 118–119] ou [10, p. 29]), donde o sistema é possível e determinado e basta resolver

$$P = V^{-1}d,$$

encontrando-se os coeficientes do polinómio caraterístico da matriz  $A$ .

**Exemplo 4.5** Para uma melhor compreensão do método dos coeficientes indeterminados, apresentamos agora um exemplo.

Retomemos a matriz (4.11) e, com o auxílio do MATLAB®, obtemos o seguinte:

- cálculo dos determinantes,  $D_k = \det(kI_4 - A)$ ,  $k = 0, 1, 2, 3$ ,

$$D_0 = 36,$$

$$D_1 = 151,$$

$$D_2 = 208$$

e

$$D_3 = 165;$$

- construção da matriz  $d$

$$d = \begin{bmatrix} D_0 \\ D_1 - 1 \\ D_2 - 2^4 \\ D_3 - 3^4 \end{bmatrix} = \begin{bmatrix} 36 \\ 150 \\ 192 \\ 84 \end{bmatrix};$$



- construção da matriz  $V$

$$V = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 \\ 1 & 2 & 4 & 8 \\ 1 & 3 & 9 & 27 \end{bmatrix};$$

- cálculo dos coeficientes do polinómio caraterístico

$$P = V^{-1}d = \begin{bmatrix} 36 \\ 124 \\ 3 \\ -13 \end{bmatrix}.$$

Assim, e como nos métodos anteriores, o polinómio caraterístico da matriz  $A$  é:

$$p(x) = x^4 - 13x^3 + 3x^2 + 124x + 36.$$

Em [4, pp 430] ou [27] podemos ver uma análise da eficiência de cada um dos métodos descritos neste capítulo, em termos do número de operações envolvidas na sua utilização, em função da ordem da matriz. Observamos que, nesse contexto, o melhor método é o de Danilevsky, a partir de uma ordem superior a 5.



## Capítulo 5

# Métodos Numéricos de Aproximação de Zeros de Polinómios

Neste capítulo, tentaremos fazer uma pequena revisão histórica do problema da determinação das raízes de um polinómio, ou seja, vamos relembrar alguns métodos já trabalhados. No entanto, este problema não foi abandonado pelos investigadores. Ainda hoje se escrevem artigos sobre o assunto, como por exemplo [23].

Como já referimos, o problema de determinar os valores próprios de uma matriz pode passar pelo cálculo das raízes do polinómio característico. No entanto, quando estamos perante um polinómio de grau superior a 4 sabemos que não possuímos métodos diretos de determinar as suas raízes. Para tal, teremos de utilizar métodos numéricos de aproximação de raízes de polinómios, partindo da localização destas.

Os métodos numéricos para aproximar zeros de funções não lineares mais conhecidos são os abordados em qualquer curso introdutório de Métodos Numéricos. Temos, por exemplo, o método da bissecção, o método da corda-falsa, o método da secante ou o método de Newton-Raphson. Neste trabalho, estes métodos não serão abordados, mas podem ser vistos, por exemplo, em [19, Cap. 5], [5, Cap. 2] ou em [20, Secção 6.2].

Neste capítulo, apresentaremos métodos específicos e adaptados ao caso em que as equações não lineares a resolver são polinómios. É de ter em atenção que tomaremos sempre polinómios com coeficientes reais.

Consideremos o polinómio  $p(x) = a_n x^n + a_{n-1} x^{n-1} + \dots + a_1 x + a_0$  de coeficientes reais, com  $a_n \neq 0$ . É bem conhecido que o polinómio pode ser escrito da seguinte forma

$$p(x) = a_n (x - r_1)^{m_1} (x - r_2)^{m_2} \dots (x - r_k)^{m_k} [(x - u_1)^2 + v_1]^{n_1} [(x - u_2)^2 + v_2]^{n_2} \dots [(x - u_\ell)^2 + v_\ell]^{n_\ell}, \quad (5.1)$$

onde  $r_i \in \mathbb{R}$  e  $m_i \in \mathbb{N}, i = 1, \dots, k$ ;  $u_j, v_j \in \mathbb{R}$  e  $n_j \in \mathbb{N}, j = 1, \dots, \ell$ ; com  $\sum_{i=1}^k m_i + \sum_{j=1}^\ell n_j = n$ . Nestas condições, diz-se que  $r_i$  é uma raíz real de  $p$  com multiplicidade  $m_i, i = 1, \dots, k$ , e que  $u_j \pm i v_j$  são raízes complexas conjugadas de  $p$  com multiplicidade  $n_j, j = 1, \dots, \ell$ .

Como facilmente se constata, as raízes reais, contadas as suas multiplicidades, podem

ser caracterizadas do seguinte modo:

**Proposição 5.1**  $\alpha \in \mathbb{R}$  é uma raiz do polinómio  $p(x)$  com multiplicidade  $m \in \mathbb{N}$  se e só se  $p(\alpha) = p'(\alpha) = \dots = p^{(m-1)}(\alpha) = 0 \wedge p^{(m)} \neq 0$ .

Note-se, ainda, que se  $x_0$  for uma raiz do polinómio  $p$ , o polinómio admite uma das seguintes fatorizações:

- se  $x_0 \in \mathbb{R}$ ,  $p(x) = (x - x_0)q(x)$ ;
- se  $x_0 = \alpha + i\beta$ ,  $p(x) = [(x - \alpha)^2 + \beta^2]q(x)$ .

onde  $q(x)$  é um polinómio de grau inferior a  $n$ . Deste modo, podemos continuar a procurar raízes do polinómio  $p$ , mas desta vez trabalhando sobre o polinómio  $q$ . A este processo chamamos **método da deflação**.

Encontrado um zero de  $q$  podemos reescrever este polinómio em função dessa raiz, e assim sucessivamente, até  $p$  estar fatorizado em polinómios de grau 1 e/ou 2, tal como em (5.1).

## 5.1 Método de Newton-Horner

O método de Newton-Horner<sup>1</sup> é um método que melhora, em termos de número de operações aritméticas envolvidas, o método de Newton-Raphson, pois envolve  $n$  somas e/ou subtrações e  $n$  multiplicações, por iteração. Este método aproxima as raízes reais do polinómio.

A ideia base é a utilização da fórmula iterativa do método de Newton-Raphson,

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)},$$

e do teorema, que enunciaremos de seguida, recorrendo à regra de Ruffini. Graficamente, a ideia é semelhante à do método de Newton-Raphson, que podemos ver na Figura 5.1.

O teorema que enunciaremos de seguida servirá de apoio ao método que explicaremos mais à frente e pode ser visto, assim como a sua prova que incluímos, em [3, p. 83].

**Teorema 5.1 (Método de Horner)** *Considere-se o polinómio*

$$p(x) = a_n x^n + a_{n-1} x^{n-1} + \dots + a_1 x + a_0$$

e

$$b_n = a_n.$$

Se  $b_k = a_k + b_{k+1}x_0$ , para  $k = n - 1, n - 2, \dots, 1, 0$ , então  $b_0 = p(x_0)$ .

Se  $q(x) = b_n x^{n-1} + b_{n-1} x^{n-2} + \dots + b_2 x + b_1$  então  $p(x) = (x - x_0)q(x) + b_0$ .

---

<sup>1</sup>Este método é muito prático mesmo se os cálculos forem feitos à mão e também é facilmente programável em computadores.

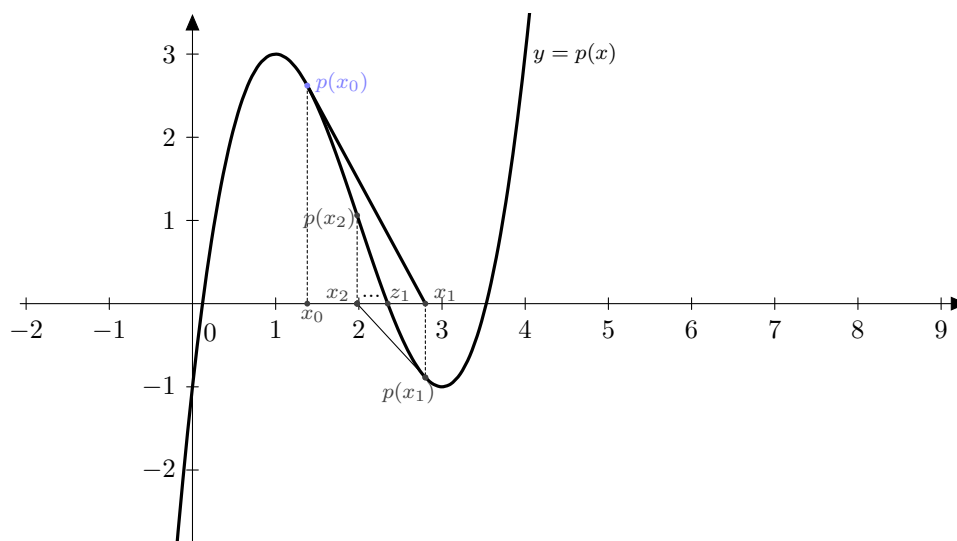


Figura 5.1: Método Iterativo de Newton-Raphson

**Prova:**

Como  $q(x) = b_n x^{n-1} + b_{n-1} x^{n-2} + \dots + b_2 x + b_1$ , temos que

$$\begin{aligned} (x - x_0)q(x) + b_0 &= (x - x_0)(b_n x^{n-1} + \dots + b_2 x + b_1) + b_0 \\ &= (b_n x^n + \dots + b_2 x^2 + b_1 x) - (b_n x_0 x^{n-1} + \dots + b_2 x_0 x + b_1 x_0) + b_0 \\ &= b_n x^n + (b_{n-1} - b_n x_0) x^{n-1} + \dots + (b_1 - b_2 x_0) x + (b_0 - b_1 x_0). \end{aligned}$$

Como, por hipótese,  $b_n = a_n$  e  $a_k = b_k - b_{k+1} x_0$ , para  $k = n-1, n-2, \dots, 1, 0$ , temos que

$$(x - x_0)q(x) + b_0 = a_n x^n + a_{n-1} x^{n-1} + \dots + a_1 x + a_0 = p(x).$$

Daqui sai imediatamente que

$$p(x_0) = b_0$$

□

Passamos a explicar o método. Seja  $p(x) = a_n x^n + a_{n-1} x^{n-1} + \dots + a_1 x + a_0$  o polinómio em estudo e  $x_0$  uma aproximação inicial a um dos zeros de  $p(x)$ . Começamos por aplicar a regra de Ruffini a  $p(x)$  em  $x = x_0$ ,

$$\begin{array}{c|ccccc} & a_n & a_{n-1} & \dots & a_1 & a_0 \\ x_0 & & a_n x_0 & \dots & b_2 x_0 & b_1 x_0 \\ \hline & a_n & b_{n-1} & \dots & b_1 & b_0 \end{array},$$

e, pelo Teorema 5.1, temos que  $p(x) = (x - x_0)q(x) + b_0$ , onde  $q(x) = b_n x^{n-1} + b_{n-1} x^{n-2} + \dots + b_1$  e  $b_0 = p(x_0)$ .

De seguida, derivamos  $p(x)$ ,

$$p'(x) = q(x) + (x - x_0)q'(x),$$

donde podemos ver que  $p'(x_0) = q(x_0)$ .

Para determinar  $p'(x_0)$  aplicamos a regra de Ruffini a  $q(x)$  em  $x = x_0$ ,

$$\begin{array}{r|rrrr} & b_n & b_{n-1} & \cdots & b_1 \\ x_0 & & b_n x_0 & \cdots & c_2 x_0 \\ \hline & b_n & c_{n-1} & \cdots & c_1 \end{array},$$

onde  $p'(x_0) = q(x_0) = c_1$ .

A última fase do processo é a aplicação da fórmula iterativa do método de Newton--Raphson para determinarmos a aproximação seguinte,  $x_1$ ,

$$x_1 = x_0 - \frac{p(x_0)}{p'(x_0)} = x_0 - \frac{b_0}{c_1}.$$

Repetimos o processo descrito anteriormente, aplicado a  $x_1$ , para encontrarmos a aproximação seguinte,  $x_2$ , e assim sucessivamente, até encontrarmos uma aproximação  $x_k$  satisfatória.

Pelo método da deflação obtemos

$$p(x) \cong (x - x_k)q_1(x).$$

Podemos recomeçar o processo descrito para encontrar uma aproximação para uma segunda raiz do polinômio  $p(x)$ , mas desta vez aplicado a  $q_1(x)$ , e assim sucessivamente, até se encontrarem todas as raízes do polinômio.

Uma das vantagens, já referida, deste método é a da diminuição do número de operações necessárias em relação ao método de Newton-Raphson, outra é a de fácil compreensão e manuseamento. No entanto, os consecutivos erros de aproximação podem levar a valores para as raízes distantes dos reais. Se a aproximação inicial não for boa, o método pode tornar-se demorado e, no caso em que as raízes são próximas, pode não funcionar tão bem.

**Exemplo 5.1** *Vamos voltar ao Exemplo 4.1 em que se chegou ao polinômio característico*

$$p(x) = x^4 - 13x^3 + 3x^2 + 124x + 36. \quad (5.2)$$

*No Exemplo A.2, do Apêndice A, verificámos que este polinômio possui 4 raízes reais e conseguimos separá-las nos intervalos  $] - 3, -2[$ ,  $] - 1, 0[$ ,  $]3, 5[$  e  $]5, 12[$ . Neste exemplo, vamos tentar aproximar o zero do intervalo  $]3, 5[$ , usando o Método de Newton-Horner.*

*Começamos com uma aproximação inicial  $x_0 = 3.5$ . Para não alongarmos o exemplo faremos apenas 4 iterações e vamos usar o comando `deconv(P,Q)` do MATLAB®, que devolve os coeficientes do polinômio resultante da divisão do polinômio  $P$  pelo polinômio  $Q$  e o resto dessa mesma divisão.*

- *Primeira iteração.*

*O polinómio resultante da divisão de  $p(x)$  por  $(x - x_0)$  é*

$$q_1(x) = x^3 - 9.5x^2 - 30.25x + 18.125$$

*e o resto dessa divisão é*

$$p(x_0) = 99.4375.$$

*O polinómio resultante da divisão de  $q(x)$  por  $(x - x_0)$  é*

$$t_1(x) = x^2 - 6x - 51.25$$

*e o resto dessa divisão é*

$$p'(x_0) = -161.25.$$

*A aproximação seguinte é*

$$\begin{aligned} x_1 &= x_0 - \frac{p(x_0)}{p'(x_0)} \\ &\cong 4.116666666666667. \end{aligned}$$

- *Segunda iteração.*

*O polinómio resultante da divisão de  $p(x)$  por  $(x - x_1)$  é*

$$q_2(x) \cong x^3 - 8.883333333333333x^2 - 33.569722222222225x - 14.195356481481497$$

*e o resto dessa divisão é*

$$p(x_1) \cong -22.437550848765504.$$

*O polinómio resultante da divisão de  $q(x)$  por  $(x - x_1)$  é*

$$t_2(x) \cong x^2 - 4.766666666666666x - 53.192500000000003$$

*e o resto dessa divisão é*

$$p'(x_1) \cong -233.1711481481482.$$

*A aproximação seguinte é*

$$\begin{aligned} x_2 &= x_1 - \frac{p(x_1)}{p'(x_1)} \\ &\cong 4.020438848486909. \end{aligned}$$

- Terceira iteração.

O polinómio resultante da divisão de  $p(x)$  por  $(x - x_2)$  é

$$q_3(x) \cong x^3 - 8.979561151513092x^2 - 33.101776495907075x - 9.083668178075669$$

e o resto dessa divisão é

$$p(x_2) \cong -0.520332429899717.$$

O polinómio resultante da divisão de  $q(x)$  por  $(x - x_2)$  é

$$t_3(x) \cong x^2 - 4.959122303026184x - 53.039624457391412$$

e o resto dessa divisão é

$$p'(x_2) \cong -495.9122303026184.$$

A aproximação seguinte é

$$\begin{aligned} x_3 &= x_2 - \frac{p(x_2)}{p'(x_2)} \\ &\cong 4.018098447992845. \end{aligned}$$

- Quarta iteração.

O polinómio resultante da divisão de  $p(x)$  por  $(x - x_3)$  é

$$q_4(x) \cong x^3 - 8.981901552007155x^2 - 33.090164686144476x - 8.959539369224757$$

e o resto dessa divisão é

$$p(x_3) \cong -0.000311234212787781.$$

O polinómio resultante da divisão de  $q(x)$  por  $(x - x_3)$  é

$$t_4(x) \cong x^2 - 4.963803104014311x - 53.035214234526443$$

e o resto dessa divisão é

$$p'(x_3) \cong -222.0602513739435.$$

A aproximação seguinte é

$$\begin{aligned} x_4 &= x_3 - \frac{p(x_3)}{p'(x_3)} \\ &\cong 4.018097046417323. \end{aligned}$$



Vimos em (3.14) que um dos zeros do polinômio  $p(x)$  é  $\lambda \cong 4.018097046416818$ , logo aplicado o Método de Newton-Horner a este polinômio, com uma aproximação inicial  $x_0 = 3.5$ , chegamos a um valor, ao fim de 4 iterações, bastante bom.

Encontrada esta raiz podemos aplicar o método da deflação. Assim, podemos reescrever o polinômio (5.2) da seguinte forma

$$p(x) \cong (x - 4.018097046417323)(x^3 - 17.0180970464173x^2 + 71.3803654778528x - 162.8132356987495)$$

e poderíamos aplicar novamente o método de Newton-Horner ao polinômio  $q(x) \cong x^3 - 17.018097x^2 + 71.380365x - 162.813236$  e encontrar uma nova raiz de  $p(x)$ .

## 5.2 Método de Müller

Aplicaremos este método apenas ao caso de raízes reais e, antes de o aplicarmos, teremos sempre de verificar a existência e unicidade de um zero num dado intervalo. Ou seja, antes de aplicarmos o Método de Müller, que iremos desenvolver de seguida, temos de provar que

$$\exists! \alpha \in [a, b] : p(\alpha) = 0.$$

Abordaremos este método seguindo [21, pp. 84–88] e [3, pp. 87–89]. O método de Müller pode ser visto como uma extensão do método da secante. Este último começa com duas aproximações,  $x_0$  e  $x_1$ , para determinar a aproximação seguinte como a interseção,  $x_2$ , da reta que passa nos pontos  $(x_0, p(x_0))$  e  $(x_1, p(x_1))$  com o eixo das abcissas. Veja-se o gráfico da Figura 5.2.

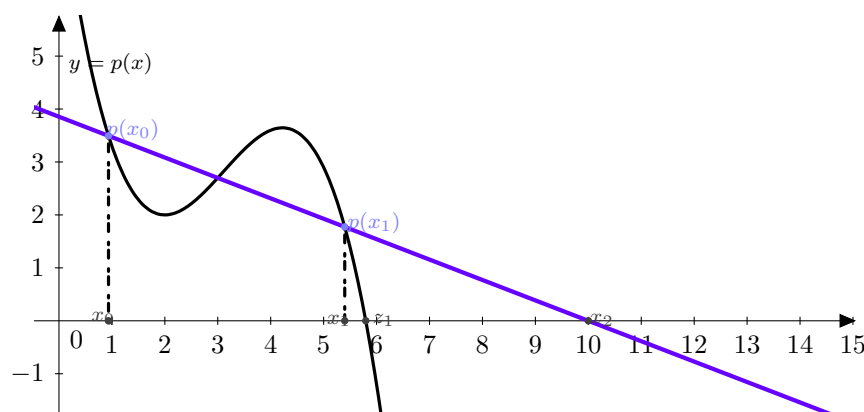


Figura 5.2: Método da Secante

O método de Müller, que iremos descrever nesta secção, utiliza três aproximações iniciais para determinar a seguinte, considerando a interseção do eixo das abcissas com a

parábola que passa nos pontos  $(x_0, p(x_0))$ ,  $(x_1, p(x_1))$  e  $(x_2, p(x_2))$ . Assim, usamos a interpolação quadrática. Veja-se o gráfico da Figura 5.3.

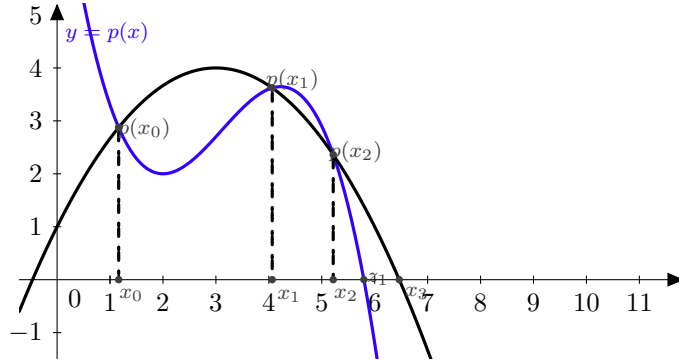


Figura 5.3: Método de Müller

Seja  $p(x) = a_n x^n + a_{n-1} x^{n-1} + \dots + a_1 x + a_0$  um polinómio com  $a_n \neq 0$ , do qual se pretende determinar as raízes e sejam  $x_0$ ,  $x_1$  e  $x_2$  três aproximações iniciais, distintas, a uma das raízes que se pretende determinar.

O método tem início com a determinação do polinómio quadrático

$$q(x) = a(x - x_2)^2 + b(x - x_2) + c,$$

que passa nos pontos  $(x_0, p(x_0))$ ,  $(x_1, p(x_1))$  e  $(x_2, p(x_2))$ . Para tal, podemos formar o seguinte sistema

$$\begin{cases} p(x_0) = a(x_0 - x_2)^2 + b(x_0 - x_2) + c \\ p(x_1) = a(x_1 - x_2)^2 + b(x_1 - x_2) + c \\ p(x_2) = a(x_2 - x_2)^2 + b(x_2 - x_2) + c \end{cases}.$$

Matricialmente, temos

$$\begin{bmatrix} p(x_0) \\ p(x_1) \\ p(x_2) \end{bmatrix} = \begin{bmatrix} (x_0 - x_2)^2 & (x_0 - x_2) & 1 \\ (x_1 - x_2)^2 & (x_1 - x_2) & 1 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} a \\ b \\ c \end{bmatrix}.$$

ou

$$P = Vd.$$

Facilmente se observa que a matriz  $V$  é de Vandermonde que, como já vimos, é invertível e, por isso, podemos resolver o sistema anterior e determinar os coeficientes  $a$ ,  $b$  e  $c$  fazendo

$$d = PV^{-1}.$$

De seguida, determinamos o zero,  $t$ , do polinómio  $q(x)$ , aplicando a fórmula resolvente, e

$$t = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}.$$

Se multiplicarmos o numerador e o denominador por  $b \pm \sqrt{b^2 - 4ac}$  obtemos

$$t = \frac{-2c}{b \pm \sqrt{b^2 - 4ac}}. \quad (5.3)$$

A aplicação desta fórmula poderá resultar em duas soluções, mas ficamos com a de menor valor absoluto, que corresponde à que não implica uma subtração no denominador (para se evitar cancelamento subtrativo). Para termos a aproximação  $x_3$  do polinómio  $p(x)$  fazemos

$$x_3 = x_2 - t = x_2 + \frac{2c}{b \pm \sqrt{b^2 - 4ac}}. \quad (5.4)$$

Determinada a aproximação  $x_3$ , repetimos o procedimento mas, desta vez, utilizando  $x_1$ ,  $x_2$  e  $x_3$  para determinarmos a aproximação  $x_4$ , e assim sucessivamente. O método termina quando encontramos um valor,  $x_k$ , satisfatório.

**Exemplo 5.2** *Vamos aplicar o Método de Müller ao polinómio (5.2). Para podermos comparar este método com o anterior, vamos aplicá-lo também ao intervalo  $]3, 5[$ . Vamos tomar como aproximações iniciais  $x_0 = 3.5, x_1 = 4$  e  $x_2 = 4.5$ . Do mesmo modo que no exemplo anterior, vamos apenas aplicar o método em 4 iterações, apenas para melhor mostrar como funciona o método.*

- *Primeira iteração:*

*Para as aproximações iniciais dadas obtemos o seguinte sistema matricial*

$$\begin{bmatrix} p(x_0) \\ p(x_1) \\ p(x_2) \end{bmatrix} = \begin{bmatrix} (x_0 - x_2)^2 & (x_0 - x_2) & 1 \\ (x_1 - x_2)^2 & (x_1 - x_2) & 1 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} a_1 \\ b_1 \\ c_1 \end{bmatrix}$$

$$\begin{bmatrix} 99.4375 \\ 04.0000 \\ -119.8125 \end{bmatrix} = \begin{bmatrix} 1 & -1 & 1 \\ 0.25 & -0.50 & 1 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} a_1 \\ b_1 \\ c_1 \end{bmatrix}$$

*donde se obtém  $a_1 = -056.7500, b_1 = -276.0000$  e  $c_1 = -119.8125$ . Aplicando a fórmula (5.4), obtemos, como valor mais próximo de  $x_2$ ,*

$$x_3 \cong 4.018158665530915.$$

- *Segunda iteração:*

*Para as aproximações  $x_1, x_2$  e  $x_3$ , obtemos o seguinte sistema matricial*

$$\begin{bmatrix} 4.0000 \\ -119.8125 \\ -0.0136833619461 \end{bmatrix} = \begin{bmatrix} 0.000329737133864 & -0.018158665530915 & 1 \\ 0.232171071602948 & 0.481841334469085 & 1 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} a_2 \\ b_2 \\ c_2 \end{bmatrix}$$

*donde se obtém  $a_2 \cong -55.1861149335077, b_2 \cong -222.0361549393026$  e  $c_2 \cong -0.0136833619461$ . Aplicando a fórmula (5.4), obtemos, como valor mais próximo de  $x_3$ ,*

$$x_4 \cong 4.018097037857959.$$

- Terceira iteração:

Para as aproximações  $x_2, x_3$  e  $x_4$ , obtemos o seguinte sistema matricial

$$\begin{bmatrix} -119.812500000000 \\ -0.0136833619461 \\ 0.0000019005816 \end{bmatrix} = \begin{bmatrix} 0.232230464921274 & 0.481902962142041 & 1 \\ 0.000000003797970 & 0.000061627672957 & 1 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} a_3 \\ b_3 \\ c_3 \end{bmatrix}$$

donde se obtém  $a_3 \cong -55.1221191771264$ ,  $b_3 \cong -222.0601966440910$  e  $c_3 \cong 0.0000019005816$ .  
Aplicando a fórmula (5.4), obtemos, como valor mais próximo de  $x_4$ ,

$$x_5 \cong 4.018097046416816.$$

- Quarta iteração:

Para as aproximações  $x_3, x_4$  e  $x_5$ , obtemos o seguinte sistema matricial

$$\begin{bmatrix} -0.013683361946057 \\ 0.000001900581609 \\ 0.000000000000810 \end{bmatrix} = \begin{bmatrix} 0.000000003796915 & 0.000061619114099 & 1 \\ 0.000000000000000 & -0.000000008558858 & 1 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} a_4 \\ b_4 \\ c_4 \end{bmatrix}$$

donde se obtém  $a_4 \cong -56.6622505984785$ ,  $b_4 \cong -222.0601026993037$  e  $c_4 \cong 0.0000000000008$ .  
Aplicando a fórmula (5.4), obtemos, como valor mais próximo de  $x_5$ ,

$$x_6 \cong 4.018097046416816.$$

Este método devolveu, em 4 iterações, uma aproximação muito boa do valor próprio pretendido. Verificamos que apenas em 3 iterações esse valor é o mesmo. Neste caso, este método é melhor do que o método de Newton-Horner, pois dá uma melhor aproximação.

Encontrada a raiz podemos reescrever o polinômio (5.2) da seguinte forma

$$p(x) \cong (x - 4.018097046416816)(x^3 - 17.0180970464168x^2 + 71.3803654778421x - 162.8132356986704)$$

e poderíamos aplicar novamente o método ao polinômio  $q(x) \cong x^3 - 17.018097x^2 + 71.380365x - 162.813236$  e encontrar uma nova raiz de  $p(x)$ .

## 5.3 Método de Bairstow

Este é um método que tanto devolve raízes reais como raízes complexas. Apresentamos o método seguindo [21, pp. 79–84]. Este método é um método iterativo que extrai um fator quadrático do polinômio  $p(x)$ , ou seja, sendo  $p(x)$  um polinômio de grau  $n$ ,

$$p(x) = a_n x^n + a_{n-1} x^{n-1} + \cdots + a_1 x + a_0,$$

com  $a_n \neq 0$ , aplicado o método de Bairstow, iremos decompô-lo da seguinte forma

$$\begin{aligned} p(x) &= (x^2 + \alpha x + \beta)(b_n x^{n-2} + b_{n-1} x^{n-3} + \cdots + b_3 x + b_2) + r(x) \\ &= d(x)q(x) + r(x) \end{aligned} \tag{5.5}$$

onde  $r(x) = b_1x + b_0$  é o resto da divisão entre  $p(x)$  e  $d(x) = (x^2 + \alpha x + \beta)$ . Do fator quadrático,  $d(x)$ , poderemos extrair duas raízes do polinômio, que poderão ser reais, duas simples ou uma com multiplicidade 2, ou um par de raízes complexas conjugadas.

Se multiplicarmos o termo quadrático pelo polinômio  $q(x)$ , obtemos para os coeficientes do polinômio  $p(x)$ ,  $a_i$ ,  $i = 1, \dots, n$ ,

$$\begin{aligned} a_n &= b_n \\ a_{n-1} &= b_{n-1} + \alpha b_n \\ a_{n-2} &= b_{n-2} + \alpha b_{n-1} + \beta b_n \\ &\vdots \\ a_2 &= b_2 + \alpha b_3 + \beta b_4 \\ a_1 &= b_1 + \alpha b_2 + \beta b_3 \\ a_0 &= b_0 + \beta b_2 \end{aligned} \tag{5.6}$$

Das equações (5.6) podemos extrair os coeficientes  $b_i$ ,  $i = 1, \dots, n$ , do polinômio  $q(x)$

$$\begin{aligned} b_n &= a_n \\ b_{n-1} &= a_{n-1} - \alpha b_n \\ b_{n-2} &= a_{n-2} - \alpha b_{n-1} - \beta b_n \\ &\vdots \\ b_2 &= a_2 - \alpha b_3 - \beta b_4 \\ b_1 &= a_1 - \alpha b_2 - \beta b_3 \\ b_0 &= a_0 - \beta b_2. \end{aligned} \tag{5.7}$$

Se exigirmos que  $d(x)$  seja um fator quadrático exato na fatorização do polinômio  $p(x)$  então teremos de exigir que  $r(x) = 0$ .

O objetivo do método de Bairstow é encontrar o fator  $(x^2 + \alpha x + \beta)$  de tal forma que  $r(x) = b_1x + b_0 = 0$ , ou seja, pretende encontrar o fator  $(x^2 + \alpha x + \beta)$  tal que  $b_1$  e  $b_0$  são simultaneamente nulos.

Como se pode observar nas equações (5.7),  $b_1$  e  $b_0$  dependem dos valores  $\alpha$  e  $\beta$ , então se denotarmos por  $\bar{\alpha}$  e  $\bar{\beta}$  os valores exatos de  $\alpha$  e  $\beta$  obtemos

$$\begin{cases} b_0(\bar{\alpha}, \bar{\beta}) = 0 \\ b_1(\bar{\alpha}, \bar{\beta}) = 0 \end{cases}.$$

Este sistema é um sistema não linear de duas equações a duas incógnitas e será resolvido por um processo iterativo que corresponde ao método de Newton.

Faça-se  $\alpha$  e  $\beta$  as aproximações iniciais de  $\bar{\alpha}$  e  $\bar{\beta}$ . A aproximação seguinte pode ser dada como  $(\alpha + m\alpha)$  e  $(\beta + m\beta)$ , o que leva a que  $b_1$  e  $b_0$  se aproximem mais de zero. Assim,

aplicando séries de Taylor,

$$\begin{aligned} b_0(\alpha + m\alpha, \beta + m\beta) &= b_0(\alpha, \beta) + \Delta\alpha \left( \frac{\partial b_0}{\partial \alpha} \right) + \Delta\beta \left( \frac{\partial b_0}{\partial \beta} \right) \\ b_1(\alpha + m\alpha, \beta + m\beta) &= b_1(\alpha, \beta) + \Delta\alpha \left( \frac{\partial b_1}{\partial \alpha} \right) + \Delta\beta \left( \frac{\partial b_1}{\partial \beta} \right) \end{aligned} \quad (5.8)$$

onde as derivadas parciais são avaliadas em  $\alpha$  e  $\beta$  e  $\Delta\alpha$  e  $\Delta\beta$  são tais que

$$\begin{aligned} \Delta\alpha &= \bar{\alpha} - \alpha \\ \Delta\beta &= \bar{\beta} - \beta. \end{aligned}$$

Igualando as equações (5.8) a zero e assumindo que  $\alpha + m\alpha \cong \bar{\alpha}$  e  $\beta + m\beta \cong \bar{\beta}$  temos

$$\begin{cases} \Delta\alpha \left( \frac{\partial b_0}{\partial \alpha} \right) + \Delta\beta \left( \frac{\partial b_0}{\partial \beta} \right) = -b_0(\alpha, \beta) \\ \Delta\alpha \left( \frac{\partial b_1}{\partial \alpha} \right) + \Delta\beta \left( \frac{\partial b_1}{\partial \beta} \right) = -b_1(\alpha, \beta) \end{cases}. \quad (5.9)$$

O cálculo das derivadas parciais de  $b_0$  e  $b_1$ , no sistema (5.9), requer o cálculo recursivo das derivadas parciais de todos os  $b_i$ ,  $i = n, n-1, \dots, 1, 0$ , nas equações (5.7)

$$\begin{aligned} \frac{\partial b_n}{\partial \alpha} &= 0; \\ \frac{\partial b_n}{\partial \beta} &= 0; \\ \frac{\partial b_{n-1}}{\partial \alpha} &= -b_n - \alpha \frac{\partial b_n}{\partial \alpha} = -b_n; \\ \frac{\partial b_{n-1}}{\partial \beta} &= -\alpha \frac{\partial b_n}{\partial \beta} = 0; \\ \frac{\partial b_{n-2}}{\partial \alpha} &= -b_{n-1} - \alpha \frac{\partial b_{n-1}}{\partial \alpha} - \beta \frac{\partial b_n}{\partial \alpha} = -b_{n-1} + \alpha b_n; \\ \frac{\partial b_{n-2}}{\partial \beta} &= -\alpha \frac{\partial b_{n-1}}{\partial \beta} - b_n - \beta \frac{\partial b_n}{\partial \beta} = -b_n; \\ &\vdots \\ \frac{\partial b_2}{\partial \alpha} &= -b_3 - \alpha \frac{\partial b_3}{\partial \alpha} - \beta \frac{\partial b_4}{\partial \alpha}; \\ \frac{\partial b_2}{\partial \beta} &= -\alpha \frac{\partial b_3}{\partial \beta} - b_4 - \beta \frac{\partial b_4}{\partial \beta}; \\ \frac{\partial b_1}{\partial \alpha} &= -b_2 - \alpha \frac{\partial b_2}{\partial \alpha} - \beta \frac{\partial b_3}{\partial \alpha}; \\ \frac{\partial b_1}{\partial \beta} &= -\alpha \frac{\partial b_2}{\partial \beta} - b_3 - \beta \frac{\partial b_3}{\partial \beta}; \\ \frac{\partial b_0}{\partial \alpha} &= -\beta \frac{\partial b_2}{\partial \alpha}; \\ \frac{\partial b_0}{\partial \beta} &= -b_2 - \beta \frac{\partial b_2}{\partial \beta}. \end{aligned} \quad (5.10)$$

Todas as derivadas parciais são avaliadas em  $(\alpha, \beta)$  e, depois de calculadas, podemos resolver as equações (5.9), em função de  $\Delta\alpha$  e  $\Delta\beta$

$$\begin{cases} \Delta\alpha = \frac{b_1 \left( \frac{\partial b_0}{\partial \beta} \right) - b_0 \left( \frac{\partial b_1}{\partial \beta} \right)}{\frac{\partial b_0}{\partial \alpha} \frac{\partial b_1}{\partial \beta} - \frac{\partial b_1}{\partial \alpha} \frac{\partial b_0}{\partial \beta}} \\ \Delta\beta = \frac{b_0 \left( \frac{\partial b_1}{\partial \alpha} \right) - b_1 \left( \frac{\partial b_0}{\partial \alpha} \right)}{\frac{\partial b_1}{\partial \beta} \frac{\partial b_0}{\partial \alpha} - \frac{\partial b_0}{\partial \beta} \frac{\partial b_1}{\partial \alpha}} \end{cases}. \quad (5.11)$$

Como  $\Delta\alpha$  e  $\Delta\beta$  são conhecidos podemos calcular novas aproximações para  $\bar{\alpha}$  e  $\bar{\beta}$  fazendo

$$\bar{\alpha} \leftarrow \alpha + \Delta\alpha$$

e

$$\bar{\beta} \leftarrow \beta + \Delta\beta.$$

Falta-nos mostrar como poderemos escolher as aproximações iniciais de  $\bar{\alpha}$  e  $\bar{\beta}$ . Retornemos a decomposição de  $p(x)$ ,

$$p(x) = d(x)q(x) + r(x)$$

onde  $d(x) = (x^2 + \alpha x + \beta)$ .

Podem ocorrer três situações.

- Suponhamos que conhecemos duas aproximações de raízes,  $r_1$  e  $r_2$ , reais, distintas,

$$p(x) \cong (x - r_1)(x - r_2)q(x).$$

Daqui podemos ver que

$$\begin{aligned} d(x) &\cong (x - r_1)(x - r_2) \\ &\cong x^2 - (r_1 + r_2)x + r_1r_2 \end{aligned}$$

donde

$$\begin{cases} \alpha &= -(r_1 + r_2) \\ \beta &= r_1r_2 \end{cases},$$

podem ser consideradas aproximações os valores exatos de  $\bar{\alpha}$  e  $\bar{\beta}$ ;

- Suponhamos que partimos de um par de raízes complexas conjugadas,  $u \pm iv$ , aproximadas

$$p(x) \cong [x - (u + iv)][x - (u - iv)]q(x).$$

Daqui podemos ver que

$$\begin{aligned} d(x) &\cong [x - (u + iv)][x - (u - iv)] \\ &\cong x^2 - 2ux + (u^2 + v^2) \end{aligned}$$

donde

$$\begin{cases} \alpha &= -2u \\ \beta &= u^2 + v^2 \end{cases},$$

podem ser consideradas aproximações os valores exatos de  $\bar{\alpha}$  e  $\bar{\beta}$ ;

- Suponhamos que temos uma aproximação de uma raiz real dupla,  $r$ ,

$$p(x) \cong (x - r)^2 q(x).$$

Daqui podemos ver que

$$\begin{aligned} d(x) &\cong (x - r)^2 \\ &\cong x^2 - 2rx + r^2 \end{aligned}$$

donde

$$\begin{cases} \alpha &= -2r \\ \beta &= r^2 \end{cases},$$

podem ser consideradas aproximações os valores exatos de  $\bar{\alpha}$  e  $\bar{\beta}$ .

A aplicação do método de Bairstow pode ser resumida da seguinte forma:

1. escolher as aproximações iniciais  $\alpha$  e  $\beta$ , de  $\bar{\alpha}$  e  $\bar{\beta}$ , respetivamente;
2. usar as equações (5.7) para calcular  $b_1$  e  $b_0$ ;
3. usar as equações (5.10) para calcular as derivadas parciais de  $b_1$  e  $b_0$ , em relação a  $\alpha$  e  $\beta$ ;
4. resolver as equações (5.11);
5. calcular as novas aproximações de  $\bar{\alpha}$  e  $\bar{\beta}$  como  $\bar{\alpha} \leftarrow \alpha + \Delta\alpha$  e  $\bar{\beta} \leftarrow \beta + \Delta\beta$ , respetivamente;
6. usar as novas aproximações e repetir o processo do ponto 2 ao 5 até obter resultados satisfatórios, por exemplo  $\|[\Delta\alpha, \Delta\beta]\| < \text{tol}$ , sendo  $\text{tol}$  um valor de tolerância pré-definido;
7. usar as últimas aproximações obtidas,  $\alpha$  e  $\beta$ , e calcular as raízes do polinômio  $p(x)$  para o fator quadrático, usando a fórmula resolvente para equações do segundo grau, ou seja,

$$x_{1,2} = \frac{-\alpha \pm \sqrt{\alpha^2 - 4\beta}}{2}.$$

Agora que temos

$$p(x) \cong (x^2 + \alpha x + \beta)q(x),$$

com  $q(x)$  um polinômio de grau  $(n - 2)$ , aplicamos o método de Bairstow, descrito acima, ao polinômio  $q(x)$  e obtemos mais um fator quadrático de  $p(x)$ , e assim sucessivamente, até encontrarmos todas as raízes do polinômio em estudo.

O método descrito consegue encontrar qualquer tipo de raiz, seja real ou complexa, contando multiplicidades. No entanto, apenas é aplicável a polinômios de coeficientes reais e depende de uma boa escolha das aproximações iniciais de  $\alpha$  e  $\beta$  e, devido a possíveis erros de arredondamento, as raízes encontradas podem não ser muito precisas. O método, quando converge, converge quadraticamente ([29, pp. 449–451] ou [21, pp. 79–84]).



**Exemplo 5.3** Vamos aplicar o Método de Bairstow ao polinômio (5.2). Como já referimos, no Exemplo A.2, do Apêndice A, verificamos que este polinômio possui 4 raízes reais e que se dividem pelos intervalos  $] - 3, -2[$ ,  $] - 1, 0[$ ,  $]3, 5[$  e  $]5, 12[$ . De modo a aproximar as raízes dos dois primeiros intervalos, tomemos como aproximações iniciais

$$\alpha = -(-2.5 - 0.5) = 3$$

e

$$\beta = (-2.5) \times (-0.5) = 1.25.$$

Aplicado o método, obtemos, na primeira iteração,

$$\alpha \cong 2.864908854166667,$$

$$\beta \cong 0.776204427083333,$$

e como resto da divisão do polinômio  $p(x)$  por  $(x^2 + \alpha x + \beta)$

$$r_1(x) \cong -0.271015053580470x + -1.005789339007734.$$

Continuamos o processo e, na quinta iteração,

$$\alpha \cong 2.858571240005783,$$

$$\beta \cong 0.756681268828994,$$

e como resto da divisão do polinômio  $p(x)$  por  $(x^2 + \alpha x + \beta)$

$$r_5(x) \cong 0.$$

Então podemos afirmar que um fator quadrático do polinômio em estudo é, aproximadamente,

$$x^2 + 2.858571240005783x + 0.756681268828994,$$

donde podemos extrair duas raízes do polinômio, sendo elas

$$\lambda_1 \cong -0.295188571810782$$

$$\lambda_2 \cong -2.563382668195001.$$

As aproximações obtidas, depois de aplicado este método, são muito boas.

Agora, podemos reescrever o polinômio em estudo da seguinte forma

$$p(x) \cong (x^2 + 2.858571240005783x + 0.756681268828994)(x^2 - 15.858571240005784x + 47.576174385434392)$$

donde de  $q(x) = x^2 - 15.858571240005784x + 47.576174385434392$  podemos extrair, facilmente, as outras duas raízes,

$$\lambda_3 \cong 11.840474193588964$$

$$\lambda_4 \cong 4.018097046416820.$$



## Capítulo 6

# Métodos Matriciais Iterativos de Aproximação de Valores Próprios

Uma outra forma de obter aproximações dos valores próprios de uma matriz é usando métodos matriciais iterativos. Neste capítulo, iremos abordar o Método da Potência, o Método QR e o Método de Jacobi. No entanto, existem outros métodos de cálculo iterativo de valores próprios de uma dada matriz que não abordaremos neste trabalho, como por exemplo o método de Lanczos. Alguns autores reservam capítulos dos seus trabalhos a este tema. Deixamos aqui alguns, [19, Cap. 7], [20, Cap. 5], [22, Cap. 10], [8, Cap. 7–9] ou [11, Cap. 7].

### 6.1 O Método da Potência

O Método da Potência, ou o Método das Potências Diretas, aproxima o valor próprio dominante, ou seja, dá-nos uma aproximação do valor próprio de maior valor absoluto, caso a matriz seja diagonalizável. Na prática, o cálculo do valor próprio dominante é utilizado na Estabilidade dos Sistemas Lineares Discretos ([9]) ou nos Algoritmos de Page Ranking ([18] ou [6]).

Vamos descrever o método na perspectiva de dois autores, Quarteroni ([20, pp. 192–199]) e Demidovich ([4, pp.430–436]), descrevendo a forma como cada um aborda o método. É de referir que o núcleo central do método, razão da sua designação, é comum às duas abordagens. Começaremos por descrever esse núcleo e iremos mostrar como os autores abordam a questão do cálculo do valor próprio dominante.

Considere-se a matriz  $A \in \mathbb{C}^{n \times n}$  diagonalizável. Tomemos os vetores próprios de  $A$ ,  $x_1, \dots, x_n$  associados aos valores próprios  $\lambda_1, \dots, \lambda_n$ , respetivamente.

Suponhamos que  $\lambda_1$  possui multiplicidade simples e que podemos ordenar os valores próprios de  $A$  da seguinte forma:

$$|\lambda_1| > |\lambda_2| \geq |\lambda_3| \geq \dots \geq |\lambda_n|. \quad (6.1)$$

Seja  $y^{(0)} \in \mathbb{C}^n$  um vetor qualquer, não nulo. O Método das Potências Diretas é um método iterativo que, em cada iteração, calcula o vetor  $y^{(k)}$  da seguinte forma:

$$y^{(k)} = \frac{Ay^{(k-1)}}{\|Ay^{(k-1)}\|_2}, \quad k \in \mathbb{N}. \quad (6.2)$$

Note-se que  $y^{(k)}$  é uma sucessão de vetores normalizados que, como veremos a seguir, converge para o vetor próprio dominante,  $x_1$ . Este é o âmago do método para os dois autores, podendo considerar-se o vetor  $y^{(k)}$  normalizado, ou não.

No entanto, para a determinação do valor próprio dominante os processos começam a diferenciar-se. Deste modo, em [20, p. 192], é usada a sucessão de escalares

$$\lambda^{(k)} = (y^{(k)})^* Ay^{(k)}, \quad k \in \mathbb{N}, \quad (6.3)$$

que tenderá para o valor próprio de maior valor absoluto,  $\lambda_1$ .

Por outro lado, Demidovich tem em atenção a noção de colinearidade entre vetores<sup>1</sup>. Tomando a sucessão de vetores não normalizados,  $y^{(k)} = Ay^{(k-1)}$ , o objetivo das iterações é ir encontrando vetores  $y^{(k)}$  cuja direção se vá alinhando com a direção do vetor próprio dominante  $x_1$ . Assim, tomando as  $j$ -ésimas coordenadas dos vetores  $y^{(k+1)}$  e  $y^{(k)}$  o valor próprio dominante será dado por

$$\lim_{k \rightarrow \infty} \frac{y_j^{(k+1)}}{y_j^{(k)}} = \lambda_1. \quad (6.4)$$

Estamos agora em condições para mostrar como é que cada autor explica a convergência do método.

Em qualquer das abordagens o método inicia-se com  $y^{(0)}$ , vetor não nulo arbitrário. Admitindo que a matriz  $A$  é diagonalizável, o vetor  $y^{(0)}$  pode escrever-se como combinação linear de uma base de vetores próprios de  $A$ , isto é, existem  $x_1, x_2, \dots, x_n$  vetores próprios de  $A$  tais que

$$y^{(0)} = \sum_{i=1}^n c_i x_i, \quad (6.5)$$

com  $c_i \in \mathbb{C}$ ,  $i = 1, \dots, n$ .

No estudo da convergência das duas variantes do método, vamos supor  $c_1 \neq 0$ .

O teorema que se segue demonstra que o método das potências usando a iteração (6.3) converge. A prova apresentada reflete o tratamento apresentado em [20, pp. 193–194].

**Teorema 6.1** *Seja  $A \in \mathbb{C}^{n \times n}$  uma matriz diagonalizável cujos valores próprios são tais que*

$$|\lambda_1| > |\lambda_2| \geq |\lambda_3| \geq \dots \geq |\lambda_n|. \quad (6.6)$$

*Tomando as igualdades (6.2), (6.3) e (6.5), e supondo  $c_1 \neq 0$  temos que*

---

<sup>1</sup>Recordamos que dois vetores  $\vec{u}$  e  $\vec{v}$  são colineares se existir um escalar  $\alpha$  tal que  $\vec{u} = \alpha \vec{v}$ .

- i) o vetor  $y^{(k)}$  alinha-se com a direção de  $x_1$ , quando  $k \rightarrow \infty$ ;  
 ii)  $\lim_{k \rightarrow \infty} \lambda^{(k)} = \lambda_1$ .

**Prova:**

- i) Atendendo a (6.2) e (6.5) podemos escrever  $y^{(1)}$  da seguinte forma

$$y^{(1)} = \frac{\sum_{i=1}^n c_i A x_i}{\left\| \sum_{i=1}^n c_i A x_i \right\|_2}.$$

Como  $x_i$  são vetores próprios associados a  $\lambda_i$ , respetivamente para  $i = 1, \dots, n$ , temos

$$y^{(1)} = \frac{\sum_{i=1}^n c_i \lambda_i x_i}{\left\| \sum_{i=1}^n c_i \lambda_i x_i \right\|_2}.$$

Repetindo este raciocínio podemos escrever a igualdade (6.2) em função de  $y^{(0)}$

$$\begin{aligned} y^{(k)} &= \frac{A y^{(k-1)}}{\left\| A y^{(k-1)} \right\|_2} \\ &= \frac{A^k y^{(0)}}{\left\| A^k y^{(0)} \right\|_2}. \end{aligned} \quad (6.7)$$

Note-se que

$$\begin{aligned} A^k y^{(0)} &= \sum_{i=1}^n c_i \lambda_i^k x_i \\ &= c_1 \lambda_1^k x_1 + \sum_{i=2}^n c_i \lambda_i^k x_i. \end{aligned}$$

Se  $c_1 \neq 0$  ( $\lambda_1 \neq 0$ )

$$\begin{aligned} A^k y^{(0)} &= c_1 \lambda_1^k \left( x_1 + \sum_{i=2}^n \frac{c_i}{c_1} \left( \frac{\lambda_i}{\lambda_1} \right)^k x_i \right) \\ &= c_1 \lambda_1^k (x_1 + z^{(k)}), \end{aligned} \quad (6.8)$$

onde  $z^{(k)} = \sum_{i=2}^n \frac{c_i}{c_1} \left( \frac{\lambda_i}{\lambda_1} \right)^k x_i$ . Como, por (6.1),

$$\lim_{k \rightarrow \infty} \left( \frac{\lambda_i}{\lambda_1} \right)^k = 0,$$

então  $z^{(k)} \rightarrow 0$ , quando  $k \rightarrow \infty$ .

De (6.8) conclui-se, de forma intuitiva, que à medida que  $k$  cresce o vetor  $A^k y^{(0)}$  alinha-se com  $x_1$  e, portanto, o mesmo acontece com  $y^{(k)}$ , que é a sua versão normalizada. Mais rigorosamente,

$$\begin{aligned} y^{(k)} &= \frac{c_1 \lambda_1^k (x_1 + z^{(k)})}{\|c_1 \lambda_1^k (x_1 + z^{(k)})\|_2} \\ &= \frac{c_1 \lambda_1^k (x_1 + z^{(k)})}{|c_1 \lambda_1^k| \|x_1 + z^{(k)}\|_2}. \end{aligned}$$

Isto é,

$$y^{(k)} = \sigma_k \frac{x_1 + z^{(k)}}{\|x_1 + z^{(k)}\|_2},$$

onde

$$\sigma_k = \frac{c_1 \lambda_1^k}{|c_1 \lambda_1^k|} = \pm 1 \quad (6.9)$$

Como

$$\lim_{k \rightarrow \infty} \frac{x_1 + z^{(k)}}{\|x_1 + z^{(k)}\|_2} = \frac{x_1}{\|x_1\|_2},$$

o vetor  $y^{(k)}$  alinha-se com a direção de  $x_1$ , quando  $k \rightarrow \infty$ .

ii) Seja

$$\tilde{y}^{(k)} = A^k y^{(0)}$$

e considere-se

$$\tilde{q}^{(k)} = \frac{\tilde{y}^{(k)}}{c_1 \lambda_1^k}.$$

Então, por (6.8),

$$\tilde{q}^{(k)} = x_1 + z^{(k)}.$$

Assim,

$$\begin{aligned} \|\tilde{q}^{(k)} - x_1\|_2 &= \|z^{(k)}\|_2 \\ &= \left\| \sum_{i=2}^n \frac{c_i}{c_1} \left( \frac{\lambda_i}{\lambda_1} \right)^k x_i \right\|_2 \\ &\leq \sqrt{\sum_{i=1}^n \left( \frac{c_i}{c_1} \right)^2 \left( \frac{\lambda_i}{\lambda_1} \right)^{2k}} \\ &\leq \left| \frac{\lambda_2}{\lambda_1} \right|^k \sqrt{\sum_{i=1}^n \left( \frac{c_i}{c_1} \right)^2} \\ &= \left| \frac{\lambda_2}{\lambda_1} \right|^k \sqrt{C}, \end{aligned}$$

onde

$$C = \sum_{i=1}^n \left( \frac{c_i}{c_1} \right)^2.$$

Assim,

$$\lim_{k \rightarrow \infty} \tilde{q}^{(k)} = x_1$$

e

$$\lim_{k \rightarrow \infty} \frac{(\tilde{q}^{(k)})^* A \tilde{q}^{(k)}}{\|\tilde{q}^{(k)}\|_2} = \lambda_1.$$

Mas

$$\begin{aligned} \frac{\frac{(\tilde{q}^{(k)})^*}{c_1 \lambda_1^k} A \frac{\tilde{q}^{(k)}}{c_1 \lambda_1^k}}{\frac{1}{|c_1 \lambda_1^k|^2} \|\tilde{y}^{(k)}\|_2^2} &= \frac{(\tilde{y}^{(k)})^* A \tilde{y}^{(k)}}{\|\tilde{y}^{(k)}\|_2^2} \\ &= (y^{(k)})^* A y^{(k)}, \end{aligned}$$

o que nos conduz a

$$\lim_{k \rightarrow \infty} (\tilde{y}^{(k)})^* A \tilde{y}^{(k)} = \lambda_1.$$

O que mostra que a sequência  $\lambda^{(k)}$ , definida em (6.3), tende para  $\lambda_1$ , quando  $k$  tende para  $+\infty$ .

□

Em seguida, apresentamos um resultado relativo à convergência da variante do método da potência proposta pelo autor Demidovich, [4, pp. 431–433]. Recordamos que nesta variante

$$y^{(0)} = \sum_{i=1}^n c_i x_i, \text{ com } c_1 \neq 0 \quad (6.10)$$

$$y^{(k)} = A y^{(k-1)} \quad (6.11)$$

e  $y_j^{(k)}$  denota a componente  $j$  do vetor  $y^{(k)}$ , com  $j = 1, \dots, n$ .

**Teorema 6.2** *Seja  $A \in \mathbb{C}^{n \times n}$  uma matriz diagonalizável cujos valores próprios satisfazem*

$$|\lambda_1| > |\lambda_2| \geq |\lambda_3| \geq \dots \geq |\lambda_n|. \quad (6.12)$$

*Considerando a sucessão definida por (6.11), (6.10), supondo que  $c_1 \neq 0$  e que  $y_j^{(k)} \neq 0$ ,*

$$\lim_{k \rightarrow \infty} \frac{y_j^{(k+1)}}{y_j^{(k)}} = \lambda_1, \quad (6.13)$$

*com  $j = 1, \dots, n$ .*

**Prova:** Começemos por escrever  $y^{(0)}$  como combinação linear dos vetores próprios de  $A$ . Assim,

$$y^{(k)} = A^k y^{(0)} = \sum_{i=1}^n c_i \lambda_i^k x_i.$$

No espaço vetorial  $\mathbb{C}$ , escolha-se uma qualquer base (não obrigatoriamente a base unitária)  $v_1, \dots, v_n$ . Então podemos escrever os vetores próprios de  $A$  como combinação linear dos vetores desta base. Assim, temos

$$x_i = \sum_{j=1}^n x_{ij} v_j$$

e então podemos reescrever  $y^{(k)}$  da seguinte forma

$$\begin{aligned} y^{(k)} &= \sum_{i=1}^n c_i \lambda_i^k \sum_{j=1}^n x_{ij} v_j \\ &= \sum_{i=1}^n c_i \lambda_i^k (x_{i1} v_1 + x_{i2} v_2 + \dots + x_{in} v_n) \\ &= \sum_{i=1}^n (c_i \lambda_i^k x_{i1} v_1 + c_i \lambda_i^k x_{i2} v_2 + \dots + c_i \lambda_i^k x_{in} v_n) \\ &= \left( \sum_{i=1}^n c_i \lambda_i^k x_{i1} \right) v_1 + \left( \sum_{i=1}^n c_i \lambda_i^k x_{i2} \right) v_2 + \dots + \left( \sum_{i=1}^n c_i \lambda_i^k x_{in} \right) v_n, \end{aligned}$$

em que  $\sum_{i=1}^n c_i \lambda_i^k x_{ij}$ , para  $j = 1, \dots, n$ , é a componente  $j$  do vetor  $y^{(k)}$ .

Para a componente  $j$ , com  $j = 1, \dots, n$ , tomemos

$$\frac{y_j^{(k+1)}}{y_j^{(k)}} = \frac{\sum_{i=1}^n c_i \lambda_i^{k+1} x_{ij}}{\sum_{i=1}^n c_i \lambda_i^k x_{ij}} \quad (6.14)$$

$$= \frac{c_1 \lambda_1^{k+1} x_{1j} + \dots + c_n \lambda_n^{k+1} x_{nj}}{c_1 \lambda_1^k x_{1j} + \dots + c_n \lambda_n^k x_{nj}}. \quad (6.15)$$

Supondo, mais uma vez,  $c_1 \neq 0$ ,

$$\frac{y_j^{(k+1)}}{y_j^{(k)}} = \lambda_1 \frac{1 + \frac{c_2 x_{2j}}{c_1 x_{1j}} \left( \frac{\lambda_2}{\lambda_1} \right)^{k+1} + \dots + \frac{c_n x_{nj}}{c_1 x_{1j}} \left( \frac{\lambda_n}{\lambda_1} \right)^{k+1}}{1 + \frac{c_2 x_{2j}}{c_1 x_{1j}} \left( \frac{\lambda_2}{\lambda_1} \right)^k + \dots + \frac{c_n x_{nj}}{c_1 x_{1j}} \left( \frac{\lambda_n}{\lambda_1} \right)^k}.$$

Tendo em conta (6.12), podemos ver que

$$\lim_{k \rightarrow \infty} \left( \frac{\lambda_i}{\lambda_1} \right)^k = 0, \text{ com } i > 1,$$



donde

$$\lim_{k \rightarrow \infty} \frac{y_j^{(k+1)}}{y_j^{(k)}} = \lambda_1, \quad (6.16)$$

para  $j = 1, \dots, n$ .  $\square$

Na prática, e usando este método para calcularmos o valor próprio de maior valor absoluto, podemos tomar a média aritmética das razões  $\frac{y_j^{(k+1)}}{y_j^{(k)}}$ .

**Exemplo 6.1** *Tendo em conta a matriz, em estudo, do Exemplo (4.1), vamos determinar uma aproximação para o seu valor próprio de maior valor absoluto, aplicando o Método das Potências Diretas. Tomemos para vetor inicial*

$$y^{(0)} = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}. \quad (6.17)$$

*Vamos utilizar o método descrito por Quarteroni e o descrito por Demidovich para tentar encontrar algumas outras diferenças nos métodos.*

### Utilizando o método descrito por Quarteroni

*Vamos aplicar 10 iterações deste método. Assim, usando as fórmulas iterativas (6.2) e (6.3) temos que:*

$$\begin{aligned} y^{(1)} &\cong \begin{bmatrix} 0.390566732942472 \\ 0.520755643923295 \\ 0.650944554904119 \\ 0.390566732942472 \end{bmatrix} \\ \lambda^{(1)} &\cong 11.796610169491524 \\ &\vdots \\ y^{(9)} &\cong \begin{bmatrix} 0.431237562410942 \\ 0.511478090193881 \\ 0.663314035956336 \\ 0.335318979064420 \end{bmatrix} \\ \lambda^{(9)} &\cong 11.840474192580324 \\ y^{(10)} &\cong \begin{bmatrix} 0.431239730643622 \\ 0.511476346111769 \\ 0.663317127651600 \\ 0.335312734990189 \end{bmatrix} \\ \lambda^{(10)} &\cong 11.840474193472822. \end{aligned}$$

Em 10 iterações, encontramos para aproximação do valor próprio de maior módulo  $\lambda_1 \cong 11.840474193472822$ .

### Utilizando o método descrito por Demidovich

Tomando, agora, uma sucessão de vetores não normalizados, vamos utilizar (6.4) para calcular o valor próprio dominante. Assim, a sucessão de vetores é a seguinte:

$$\begin{aligned} y^{(1)} &= \begin{bmatrix} 9 \\ 12 \\ 15 \\ 9 \end{bmatrix} \\ y^{(2)} &= \begin{bmatrix} 117 \\ 141 \\ 177 \\ 96 \end{bmatrix} \\ &\vdots \\ y^{(9)} &= \begin{bmatrix} 3829491012 \\ 4542045777 \\ 5890384698 \\ 2977711425 \end{bmatrix} \\ y^{(10)} &= \begin{bmatrix} 45343217481 \\ 53779792422 \\ 69745273083 \\ 35256858741 \end{bmatrix}. \end{aligned}$$

O valor próprio dominante é dado por:

$$\lambda_1 \approx \frac{\frac{y_1^{(10)}}{y_1^{(9)}} + \frac{y_2^{(10)}}{y_2^{(9)}} + \frac{y_3^{(10)}}{y_3^{(9)}} + \frac{y_4^{(10)}}{y_4^{(9)}}}{4} \cong 11.840437658333006.$$

Vimos em (3.14) que o valor próprio dominante da matriz  $A$  é, aproximadamente 11.840474193588962. Comparando os valores, o método exposto por Quarteroni chega a uma melhor aproximação para o valor próprio dominante do que o método exposto por Demidovich, tendo em conta o mesmo número de iterações.

**Exemplo 6.2** Tomemos a matriz

$$B = \begin{bmatrix} 1 & 0 & -3 & 0 \\ 1 & 2 & 1 & 0 \\ 0 & 3 & 1 & -4 \\ 1 & 0 & 2 & 0 \end{bmatrix}. \quad (6.18)$$

Neste caso, o Método das Potências, em qualquer das suas variantes, não é aplicável na procura do valor próprio de maior valor absoluto. A matriz possui apenas valores próprios complexos conjugados,  $(-0.289572513005876 \pm 2.525287105704329 i)$  e  $(2.289572513005875 \pm 0.974125026043391 i)$ . e o método devolve um valor próprio real,  $\lambda \cong 0.215578434479241$ .

No ponto 3 da Subsecção 6.1.1, veremos porque é que, neste caso, o método falhou.

### 6.1.1 Acerca do Método das Potências Diretas

Depois de termos apresentado o Método das Potências Diretas convém frisar algumas das suas características antes de continuar. Algumas das observações seguintes realçam as vantagens e desvantagens do método.

1. De (6.10) podemos ver que a rapidez de convergência do vetor  $y^{(k)}$  para o vetor próprio  $x_1$ , ou seja, a rapidez de convergência do método, vai depender do valor de

$$\left| \frac{\lambda_2}{\lambda_1} \right|$$

e será tanto mais rápida quanto menor for esta razão;

2. Por vezes, a escolha do vetor inicial  $y^{(0)}$  pode não produzir o resultado pretendido, ou seja, o método não converge. Ao descrever o método, fizemos a suposição de que  $c_1 \neq 0$ . Esta suposição pode ser difícil de se garantir à partida. No entanto, como na prática  $y^{(0)}$  é um vetor arbitrário, podemos conseguir um em que a sua componente segundo  $x_1$  não seja exatamente nula. No caso de tal acontecer, os erros de arredondamento cometidos ao longo das iterações do processo asseguram um vetor  $y^{(k)}$  que não possui componente nula segundo a direção de  $x_1$ . Mesmo que tal aconteça, aconselha-se uma troca de vetor inicial. A necessidade de se trocar o vetor inicial por outro é rara mas pode acontecer e na programação do método tem de ser prevista;
3. Na descrição anterior assumimos  $\lambda_1$  com multiplicidade simples. No entanto, podem surgir casos em que a multiplicidade do valor próprio de maior módulo é superior a um. Assim, por exemplo para multiplicidade igual a 2, podemos ter ([20, p. 196]):

- $\lambda_1 = \lambda_2$  e o método ainda converge, pois de (6.8) vem que

$$A^k y^{(0)} = \lambda_1^k \left( c_1 x_1 + c_2 x_2 + \sum_{i=3}^n c_i \left( \frac{\lambda_i}{\lambda_1} \right)^k x_i \right);$$

- $\lambda_1 = -\lambda_2$  e, neste caso, os valores próprios de maior módulo são simétricos. Tendo em atenção o ponto 3 da Propriedade 2.3, aplica-se o método à matriz  $A^2$ ;

- $\lambda_1 = \overline{\lambda_2}$  e, neste caso, ocorrem oscilações na sequência de vetores  $y^{(k)}$  e o método, em geral, não converge. De facto, podemos observar que caso  $\lambda_1 \in \mathbb{C} \setminus \mathbb{R}$  não se pode garantir a existência de  $\lim_{k \rightarrow \infty} y^{(k)}$  pois já não é possível estabelecer (6.9);
4. A forma de cálculo do valor próprio dominante sugerida por Demidovich poderá falhar no caso de uma das componentes do vetor  $y^{(k)}$  ser nula o que leva a que o quociente  $\frac{y_j^{(k+1)}}{y_j^{(k)}}$  não exista, para algum  $j = 1, \dots, n$  e  $k \in \mathbb{N}$ . De acrescentar que este autor nada refere em relação a valores próprios pertencentes a  $\mathbb{C} \setminus \mathbb{R}$ ;
  5. Este é um método bastante popular e uma das suas vantagens é a do uso exclusivo da aritmética de reais. As linguagens de programação clássicas só usam aritméticas de reais, à exceção, por exemplo, do MATLAB<sup>®</sup>, sendo, por isso, um método facilmente programável usando este último. O MATLAB<sup>®</sup> permite vetorizar o código, ou seja, operar vetores e matrizes sem recorrer a operações elemento a elemento (como nas linguagens clássicas);
  6. Se a matriz  $A$  for hermítica a convergência do método é mais rápida passando a depender de  $\left| \frac{\lambda_2}{\lambda_1} \right|^2$  ([19, pp. 399–400]). Tendo em atenção as fórmulas iterativas (6.2) e (6.3), a suposição inicial (6.1),  $c_1 \neq 0$  e recordando que se a matriz  $A$  é hermítica então os seus vetores próprios formam uma base ortonormada, temos que

$$\begin{aligned}
 \lambda^{(k)} &= (y^{(k)})^* A y^{(k)}, \quad k \in \mathbb{N} \\
 &= (y^{(k)})^* y^{(k+1)} \\
 &= \sum_{i=1}^n c_i \lambda_i^k c_i \lambda_i^{k+1} \\
 &= \sum_{i=1}^n c_i^2 \lambda_i^{2k+1} \\
 &= c_1^2 \lambda_1^{2k+1} + \sum_{i=2}^n c_i^2 \lambda_i^{2k+1} \\
 &= \lambda_1 + \sum_{i=2}^n \left( \frac{c_i}{c_1} \right)^2 \frac{\lambda_i^{2k+1}}{\lambda_1^{2k}} \\
 &= \lambda_1 \left( 1 + \sum_{i=2}^n \left( \frac{c_i}{c_1} \right)^2 \left( \frac{\lambda_i}{\lambda_1} \right)^{2k+1} \right).
 \end{aligned}$$

E, assim,

$$\begin{aligned}
 \lambda^{(k)} &\leq |\lambda^{(k)}| \\
 &= \left| \lambda_1 \left( 1 + \sum_{i=2}^n \left( \frac{c_i}{c_1} \right)^2 \left( \frac{\lambda_i}{\lambda_1} \right)^{2k+1} \right) \right| \\
 &\leq |\lambda_1| \left| 1 + \sum_{i=2}^n \left( \frac{c_i}{c_1} \right)^2 \left( \frac{\lambda_i}{\lambda_1} \right)^{2k+1} \right| \\
 &\leq |\lambda_1| \left| 1 + \sum_{i=2}^n C \left| \frac{\lambda_2}{\lambda_1} \right|^{2k+1} \right|;
 \end{aligned}$$

7. Uma das desvantagens do método prende-se com o facto de no caso de um grupo de valores próprios serem muito próximos do valor próprio dominante o método poderá ter problemas de convergência;
8. O método apenas devolve o valor próprio dominante da matriz  $A$  o que, em alguns casos, poderá ser insuficiente e, portanto, é também uma desvantagem.

### 6.1.2 O Método das Potências Inversas

Como já referimos, o Método das Potências Diretas apenas devolve um valor próprio, o de maior valor absoluto. Em algumas situações poderá ser necessário o cálculo de mais alguns valores próprios ou mesmo de todos. No entanto, podemos adaptar o método das potências, descrito anteriormente, para obter qualquer outro valor próprio. A esse método chamamos **Método das Potências Inversas**.

Sendo  $A$  uma matriz diagonalizável, basta aplicar o método descrito anteriormente à matriz  $M_\mu^{-1} = (A - \mu I_n)^{-1}$ , em que  $\mu$  é um escalar dado e uma boa aproximação de um qualquer valor próprio,  $\lambda$ , da matriz  $A$ . Uma outra vantagem deste método é a de que poderá ser aplicado para refinar uma aproximação de um valor próprio encontrado por métodos não matriciais.

Vamos ver em mais pormenor como é que o método funciona.

Seja  $\mu$  uma aproximação de um valor próprio não dominante da matriz  $A$ , a calcular. Tomemos a matriz  $M_\mu = (A - \mu I_n)$ . Sabemos que  $(\lambda - \mu)$  é valor próprio de  $A$ . Se  $\mu$  for uma boa aproximação de  $\lambda$ , e este estiver suficientemente afastado dos restantes valores próprios de  $A$ , então  $(\lambda - \mu)$  é o menor valor próprio de  $M_\mu$ , em módulo. Assim,  $M_\mu^{-1}$  terá

$$\frac{1}{\lambda - \mu}$$

como o maior valor próprio, em módulo.

Deste modo, basta aplicar o Método das Potências Diretas à matriz  $M_\mu^{-1}$  obtendo  $\tilde{\lambda}_n$  como valor próprio dominante de  $M_\mu^{-1}$ . Para obtermos o valor próprio desejado, ou seja,

o valor próprio de  $A$ , basta considerarmos que

$$\begin{aligned}\frac{1}{\lambda - \mu} &= \tilde{\lambda}_n \\ \lambda - \mu &= \frac{1}{\tilde{\lambda}_n} \\ \lambda &= \frac{1}{\tilde{\lambda}_n} + \mu.\end{aligned}$$

**Exemplo 6.3** *Tomemos, novamente, a matriz do Exemplo (4.1) para a qual vamos encontrar uma aproximação para o valor próprio de menor valor absoluto. Assim, iremos aplicar o Método das Potências Inversas à matriz  $A$ , ou seja, aplicar o Método das Potências Directas à matriz  $A^{-1}$ , tomando como vetor inicial o vetor (6.17).*

*No exemplo anterior, mostrámos como se aplica o método. Neste exemplo, mostramos apenas o resultado obtido após 10 iterações. Assim, o valor próprio de menor valor absoluto é, aproximadamente:*

$$\lambda_n \cong -0.295188571810782.$$

**Exemplo 6.4** *No capítulo anterior, estudámos métodos numéricos de aproximação de polinómios os quais aplicámos ao exemplo que nos acompanha ao longo de quase todo o trabalho. Aplicado, por exemplo, o Método de Newton-Horner, encontrámos o valor aproximado  $\lambda \cong 4.018097046417323$ . Como vimos, o Método das Potências Inversas pode servir para melhorar esta aproximação. Assim, aplicando o método à matriz  $A - 4.018097046417323 I_4$  e tomando como vetor inicial o vetor (6.17) obtemos o seguinte resultado, aproximado, para o valor próprio, em 10 iterações:*

$$\lambda \cong 4.018097046416820.$$

### 6.1.3 Acerca do Método das Potências Inversas

Inverter a matriz  $M_\mu$  pode ser computacionalmente dispendioso e trabalhoso e, portanto, na prática usam-se outras abordagens como, por exemplo, resolver o seguinte sistema

$$M_\mu y^{(k)} = y^{(k-1)}$$

por um dos métodos conhecidos. Um exemplo é o da decomposição<sup>2</sup> LU que podemos ver em maior detalhe, por exemplo, em [15, pp. 61–63].

Para cada  $\mu$ , essa decomposição só será efetuada uma vez e poderá ser utilizada em todas as iterações, produzindo a matriz triangular inferior  $L$  e a matriz triangular superior  $U$  de tal forma que

$$M_\mu = LU,$$

---

<sup>2</sup> **Decomposição LU:** Seja  $A \in \mathbb{R}^{n \times n}$  uma matriz invertível. Seja  $A_k$  a matriz formada pelas primeiras  $k$  linhas e primeiras  $k$  colunas de  $A$ . Se  $\det(A_k) \neq 0$ ,  $k = 1, \dots, n-1$ , então existe uma matriz triangular inferior  $L$ , cujos elementos da diagonal principal são iguais a 1, e uma matriz triangular superior  $U$  tal que  $A = LU$ . Nestas condições tal decomposição é única ([10, p. 160]).

tendo em atenção eventuais troca de linhas. Deste modo, a determinação de  $y^{(k-1)}$  é dada da seguinte forma

$$\begin{aligned}Lw^{(k)} &= y^{(k-1)} \\Uu^{(k)} &= w^{(k)},\end{aligned}$$

sendo  $w^{(k)}$  um vetor auxiliar. O cálculo anterior envolve dois sistemas triangulares facilmente resolúveis por substituições sucessivas.

Salientamos que a decomposição anterior é realizada para cada aproximação  $\mu$  o que pode tornar o método dispendioso e, tendo em atenção que poderemos estar a trabalhar com matrizes de grandes dimensões, o método pode tornar-se moroso sendo, nestes casos, aconselhável outros métodos mais rápidos.

## 6.2 O Método QR

Como vimos, se aplicarmos o Método das Potências Diretas apenas determinarmos um valor próprio da matriz dada ou se aplicarmos o Método das Potências Inversas poderemos determinar mais valores próprios da matriz, podendo este método, no entanto, ser pouco eficiente. Nesta secção, iremos desenvolver o Método Iterativo QR que devolve aproximações de todos os valores próprios de uma dada matriz, se algumas condições de convergência forem satisfeitas. Este método é considerado muito bom, mas da mesma forma que o Método das Potências, o espectro da matriz deverá estar convenientemente separado, caso contrário poderemos ter, como no método anterior, problemas de convergência.

Em [20, pp. 215–220] podemos ver adaptações a este método (Método QR com deslocamento simples e Método QR com deslocamento duplo) de forma a minorar a lentidão de convergência. Ainda assim, estas técnicas de deslocamento, ou “*shifting*”, não garantem a convergência do método. Já foram encontradas matrizes para as quais o método com deslocamento duplo entra em ciclo (podemos ver um exemplo em [20, pp. 243–244]). Apesar de este ser um método muito utilizado, o seu estudo completo encontra-se em aberto. O artigo [26] mostra como se poderia abordar o tema do Método QR com deslocamentos pois estes são os mais utilizados na prática, falando também, em desenvolvimentos recentes neste campo.

De um modo geral, o Método QR produz, em cada iteração, uma matriz,  $T^{(k)}$ , ortogonalmente semelhante à matriz inicial que, sob certas condições, converge para uma matriz triangular (no caso de todos os valores próprios serem reais) ou para uma matriz triangular por blocos em que os blocos diagonais são  $1 \times 1$  ou  $2 \times 2$  (no caso de existirem valores próprios complexos). O processo iterativo deste método necessita, a cada iteração, da decomposição QR da matriz  $T^{(k)}$  e, por isso, começamos esta secção por tratar essa fatorização.

### 6.2.1 Fatorização QR de matrizes quadradas

Nesta secção, mostraremos três processos diferentes de obter a decomposição QR, começando pelo que utiliza o Processo de Ortonormalização de Gram-Schmidt. De seguida mostraremos como utilizar as matrizes de reflexão de Householder e, finalmente, como utilizar as matrizes de rotação de Givens, para o mesmo efeito.

**Teorema 6.3 (Fatorização QR)** *Uma matriz  $A \in \mathbb{R}^{n \times n}$  pode ser fatorizada na forma*

$$A = QR,$$

onde  $Q$  é uma matriz de colunas ortonormadas e  $R$  é uma matriz triangular superior. Caso  $A$  seja invertível a decomposição é única.

**Prova:**

- **1º Caso:** A matriz  $A$  é invertível.

Por hipótese, temos uma matriz  $A$  cujas colunas são linearmente independentes. Vamos escrever a matriz da seguinte forma

$$A = \begin{bmatrix} A_1 & A_2 & \cdots & A_n \end{bmatrix},$$

onde  $A_i$ ,  $i = 1, \dots, n$ , são os vetores coluna que formam a matriz  $A$ , vetores esses linearmente independentes. Deste modo, podemos aplicar o Método de Ortogonalização de Gram-Schmidt para, a partir dos vetores coluna de  $A$ , formar um conjunto de vetores ortogonais. Faça-se:

$$\begin{aligned} v_1 &= A_1 \\ v_2 &= A_2 - \alpha_1^{(2)} A_1 \\ v_3 &= A_3 - \alpha_1^{(3)} A_1 - \alpha_2^{(3)} A_2 \\ &\vdots \\ v_n &= A_n - \alpha_1^{(n)} A_1 - \cdots - \alpha_{n-1}^{(n)} A_{n-1} \end{aligned}$$

onde  $\alpha_i^{(k)} = \frac{A_i^T v_k}{v_k^T v_k}$ . Assim, construímos o conjunto  $\{v_1, v_2, \dots, v_n\}$  de vetores ortogonais, cada um como combinação linear dos vetores coluna  $A_1, A_2, \dots, A_n$ . Mas podemos reescrever os vetores  $A_i$ ,  $i = 1, \dots, n$ , da seguinte forma:

$$\begin{aligned} A_1 &= v_1 \\ A_2 &= v_2 + \alpha_1^{(2)} A_1 \\ A_3 &= v_3 + \alpha_1^{(3)} A_1 + \alpha_2^{(3)} A_2 \\ &\vdots \\ A_n &= v_n + \alpha_1^{(n)} A_1 + \cdots + \alpha_{n-1}^{(n)} A_{n-1}. \end{aligned}$$



Matricialmente temos

$$\begin{bmatrix} A_1 & A_2 & \cdots & A_n \end{bmatrix} = \begin{bmatrix} v_1 & v_2 & \cdots & v_n \end{bmatrix} \begin{bmatrix} 1 & \alpha_1^{(2)} & \alpha_1^{(3)} & \cdots & \alpha_1^{(n-1)} & \alpha_1^{(n)} \\ 0 & 1 & \alpha_2^{(3)} & \cdots & \alpha_2^{(n-1)} & \alpha_2^{(n)} \\ \vdots & \vdots & \ddots & \ddots & & \vdots \\ \vdots & \vdots & & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & \cdots & 1 & \alpha_{n-1}^{(n)} \\ 0 & 0 & \cdots & \cdots & 0 & 1 \end{bmatrix}$$

$$A = VZ,$$

onde  $V$  é uma matriz ortogonal e  $Z$  uma matriz triangular superior.

Tomemos a matriz diagonal

$$D = \text{diag}(q_1, q_2, \dots, q_n),$$

onde  $q_i, i = 1, \dots, n$ , representa, respetivamente, as normas dos vetores  $v_i, i = 1, \dots, n$ . Fazendo

$$\begin{aligned} A &= (VD^{-1})(DZ) \\ &= QR, \end{aligned}$$

onde  $Q$  é uma matriz de colunas ortonormadas e  $R$  é triangular superior.

- **2º Caso:** A matriz  $A$  não é invertível ([2, pp. 10–11]).

Neste caso, as colunas  $A_1, A_2, \dots, A_n$ , da matriz  $A$ , não são linearmente independentes e, por isso, introduzimos uma alteração ao processo descrito no primeiro caso.

Consideremos as primeiras  $A_1, \dots, A_k$  colunas linearmente independentes e procedemos, para estas colunas, da mesma forma que no caso anterior, obtendo os vetores ortogonais  $v_1, \dots, v_k$ . Se a coluna  $A_{k+1}$  for linearmente dependente das  $k$  colunas anteriores, tomamos  $v_{k+1} = 0$  e a  $(k+1)$ -ésima linha da matriz  $Z$  é também tomada como nula, caso contrário, procedemos como no primeiro caso.

Prosseguimos este processo até à coluna  $A_n$ , ou seja, sempre que encontrarmos uma coluna  $i$  linearmente dependente das anteriores fazemos  $v_i = 0$  e a linha correspondente,  $i$ , na matriz  $Z$  é também nula, obtendo-se a matriz

$$V = \begin{bmatrix} v_1 & \cdots & v_k & v_{k+1} & \cdots & v_n \end{bmatrix},$$

com algumas colunas nulas e a matriz  $Z$  é triangular superior, com as correspondentes linhas nulas.

Por construção, a matriz  $V$  possui colunas ortogonais, algumas delas nulas. No entanto, a matriz  $V$  não é ortogonal mas se tomarmos para as suas colunas nulas vetores de uma base do subespaço ortogonal ao subespaço gerado pelas outras colunas, obtemos o pretendido e, assim,

$$A = \tilde{V}Z,$$

onde  $\tilde{V}$  é a matriz que se obteve de  $V$  pelo processo descrito.

Neste momento, podemos passar ao processo de normalização da mesma forma que no primeiro caso, ou seja, obtemos a matriz

$$D = \text{diag}(q_1, q_2, \dots, q_n),$$

onde  $q_i, i = 1, \dots, n$ , representa, respetivamente, as normas dos vetores  $v_i$ ,  $i = 1, \dots, n$ , e fazemos

$$\begin{aligned} A &= (VD^{-1})(DZ) \\ &= QR, \end{aligned}$$

onde  $Q$  é uma matriz de colunas ortonormadas e  $R$  é uma matriz triangular superior, com algumas linhas nulas.

□

Na demonstração do Teorema 6.3 vimos que podemos utilizar o Método de Ortonormalização de Gram-Schmidt para fatorizar uma dada matriz  $A$ . No entanto, devido a erros de arredondamento ao longo do processo, este método tende a gerar vetores que vão perdendo a sua independência linear, ou seja, ao longo do método, os vetores  $v_i$  vão perdendo precisão o que pode levar à perda da ortogonalidade entre eles ([20, pp. 85–87] ou [8, pp. 230–232]). Deste modo, na prática, este método não é muito utilizado para se obter esta decomposição.

A seguir, mostramos como utilizar as matrizes de Householder e as matrizes de Givens que, nesse âmbito, são mais usadas. Começamos pelas matrizes de Householder.

As matrizes de Householder (Definição 2.16) podem ser usadas para anular componentes de um dado vetor coluna  $x \in \mathbb{R}^n$ . Assim, se pretendermos anular todas as suas componentes, exceto a  $i$ -ésima, tomamos um vetor de Householder da seguinte forma:

$$v = x \pm \|x\|_2 e_i, \tag{6.19}$$

onde  $e_i$  é o  $i$ -ésimo vetor unitário de  $\mathbb{R}^n$ . Vamos escolher  $v = x - \|x\|_2 e_i$  e, desta forma, a matriz de reflexão dependerá apenas do vetor  $x$  e a reflexão de  $x$  é

$$Px = \|x\|_2^2 e_i.$$

De facto,

$$\begin{aligned}
 Px &= \left[ I_n - 2 \frac{(x - \|x\|e_i)(x^T - \|x\|e_i^T)}{(x - \|x\|e_i)^T(x - \|x\|e_i)} \right] x \\
 &= x - 2 \frac{(x - \|x\|e_i)(x^T - \|x\|e_i^T)}{(x - \|x\|e_i)^T(x - \|x\|e_i)} x \\
 &= x - 2 \frac{(x - \|x\|e_i)(\|x\|^2 - \|x\|x_i)}{x^T x - x^T \|x\|e_i - \|x\|e_i^T x + \|x\|^2} \\
 &= x - 2 \frac{(x - \|x\|e_i)(\|x\|^2 - \|x\|x_i)}{2\|x\|^2 - 2\|x\|x_i} \\
 &= x - x + \|x\|e_i \\
 &= \|x\|e_i.
 \end{aligned}$$

**Exemplo 6.5** Tomemos o vetor  $x = [1 \ 1 \ 1 \ 1]^T$ . Pretendemos anular todas as suas componentes exceto a primeira. Assim, escolhemos  $v$  da seguinte forma

$$v = x + \|x\|_2 e_1 = [3 \ 1 \ 1 \ 1]^T$$

e a matriz de reflexão é dada por

$$P = I_4 - 2 \frac{vv^T}{\|v\|_2^2} = \begin{bmatrix} -\frac{1}{2} & -\frac{1}{2} & -\frac{1}{2} & -\frac{1}{2} \\ -\frac{1}{2} & \frac{5}{6} & -\frac{1}{6} & -\frac{1}{6} \\ -\frac{1}{2} & -\frac{1}{6} & \frac{5}{6} & -\frac{1}{6} \\ -\frac{1}{2} & -\frac{1}{6} & -\frac{1}{6} & -\frac{5}{6} \end{bmatrix}.$$

Então

$$y = Px = [-2 \ 0 \ 0 \ 0]^T.$$

Para o caso em que se pretende anular as  $k$  últimas componentes,  $k = 1, \dots, n-2$ , de um dado vetor  $x \in \mathbb{R}^n$ , a matriz de Householder será do tipo

$$P_k = \left[ \begin{array}{c|c} I_{n-k-1} & 0 \\ \hline 0 & P_{k+1} \end{array} \right]$$

onde a matriz  $P_{k+1}$  é obtida da mesma forma que na descrição dada para o caso anterior, tomando as  $(k+1)$  últimas componentes do vetor  $x$ . As componentes do vetor  $y = P_k x$  são agora da forma

$$\begin{cases} y_j = x_j & , j = 1, \dots, n-k-1 \\ y_j = \pm \|x\|_2 & , j = n-k \\ y_j = 0 & , j = n-k+1, \dots, n \end{cases}.$$

Tomemos uma qualquer matriz  $A \in \mathbb{R}^{n \times n}$ . No sentido de obter a decomposição QR de  $A$ , a estratégia consiste em aplicar, sucessivamente, matrizes de Householder de forma a anular, convenientemente, as entradas da matriz até reduzi-la à forma triangular superior. Para tal, vamos considerar a matriz  $A$  particionada da seguinte forma

$$A = \begin{bmatrix} A_1 & A_2 & \cdots & A_{n-1} & A_n \end{bmatrix},$$

com  $A_i$ ,  $i = 1, \dots, n$ , os vetores coluna que formam a matriz  $A$ .

Para se obter a matriz de reflexão  $P_1$  para o vetor coluna  $A_1$ , de forma a anular todas as suas componentes exceto a primeira, basta tomar  $e_1$  na equação (6.19). Desta forma, obtemos a matriz

$$A^{(1)} = P_1^T A$$

com uma coluna reduzida. De seguida, encontramos a matriz de Householder,  $P_2$ , que permite anular as  $(n-2)$  últimas componentes da segunda coluna da matriz  $A^{(1)}$ , obtendo a matriz

$$A^{(2)} = P_2^T A^{(1)}$$

com duas colunas reduzidas, visto que a aplicação de  $P_2^T$  à matriz  $A^{(1)}$  preserva os zeros anteriormente obtidos.

O processo continua até à  $(n-1)$ -ésima coluna da matriz  $A^{(n-2)}$  de forma a anular a sua última componente. Assim, em  $(n-1)$  passos temos a matriz  $A$  reduzida à forma triangular superior,

$$R \equiv A^{(n-1)} = P_{n-1}^T A^{(n-2)}.$$

De um modo geral, o processo é o seguinte, para anularmos as  $(n-i)$  últimas componentes da coluna  $i$  da matriz  $A^{(k)}$  determinamos a matriz de reflexão de Householder,  $P_{k+1}$ , e obtemos a matriz

$$A^{(k+1)} = P_{k+1}^T A^{(k)}$$

com  $(k+1)$  colunas reduzidas, sendo que a operação  $P_{k+1}^T A^{(k)}$  mantém inalterados os zeros anteriormente obtidos. No final, teremos

$$\begin{aligned} R \equiv A^{(n-1)} &= P_{k+1}^T \cdots P_1^T A \\ &= (P_1 \cdots P_{n-1})^T A \end{aligned}$$

onde, se fizermos  $Q = P_1 \cdots P_{n-1}$ , temos

$$R = Q^T A, \tag{6.20}$$

sendo  $R$  uma matriz triangular superior.

Pela Propriedade 2.5, como a matriz  $Q$  é produto de matrizes ortogonais também ela é ortogonal e então podemos reescrever (6.20) do seguinte modo

$$A = QR.$$

O processo que acabámos de desenvolver devolve a fatorização QR da matriz  $A$  usando, para tal, matrizes de Householder.

**Exemplo 6.6** Retomemos a matriz  $A$ , do Exemplo 4.1.

Para a primeira redução, a matriz de Householder encontrada é

$$P_1 \cong \begin{bmatrix} 0.179605302026775 & 0.359210604053550 & 0.898026510133874 & 0.179605302026775 \\ 0.359210604053550 & 0.842719293063097 & -0.393201767342258 & -0.078640353468452 \\ 0.898026510133874 & -0.393201767342258 & 0.016995581644354 & -0.196600883671129 \\ 0.179605302026775 & -0.078640353468452 & -0.196600883671129 & 0.960679823265774 \end{bmatrix}$$

e a matriz  $A^{(1)}$ , com uma linha reduzida, é

$$A^{(1)} \cong \begin{bmatrix} 5.567764362830022 & 5.567764362830022 & 7.004606779044221 & 2.873684832428398 \\ 0 & 1.437850957522001 & 3.122281002340392 & 2.179605302026775 \\ -0.000000000000000 & 0.094627393805003 & 2.805702505850981 & -1.050986744933063 \\ -0.000000000000000 & 2.218925478761001 & 0.561140501170196 & 3.589802651013387 \end{bmatrix}.$$

Na segunda redução temos,

$$P_2 \cong \begin{bmatrix} 1.000000000000000 & 0 & 0 & 0 \\ 0 & 0.543456579425616 & 0.035765793031745 & 0.838674999226649 \\ 0 & 0.035765793031745 & 0.997198093557935 & -0.065702132790569 \\ 0 & 0.838674999226649 & -0.065702132790569 & -0.540654672983551 \end{bmatrix}$$

e

$$A^{(2)} \cong \begin{bmatrix} 5.567764362830022 & 5.567764362830022 & 7.004606779044221 & 2.873684832428398 \\ -0.000000000000000 & 2.645751311064590 & 2.267786838055363 & 4.157609203101499 \\ -0.000000000000000 & 0 & 2.872643918319924 & -1.205944356747547 \\ 0.000000000000000 & 0.000000000000000 & 2.130855144454956 & -0.043811032690703 \end{bmatrix}.$$

Na última redução temos

$$P_3 \cong \begin{bmatrix} 1.000000000000000 & 0 & 0 \\ 0 & 1.000000000000000 & 0 \\ 0 & 0 & 0.803159742697526 & 0.595763734806042 \\ 0 & 0 & 0.595763734806042 & -0.803159742697526 \end{bmatrix}$$

e

$$R \equiv A^{(3)} \cong \begin{bmatrix} 5.567764362830022 & 5.567764362830022 & 7.004606779044221 & 2.873684832428398 \\ -0.000000000000000 & 2.645751311064590 & 2.267786838055363 & 4.157609203101499 \\ -0.000000000000000 & 0.000000000000000 & 3.576678169490595 & -0.994666983734416 \\ -0.000000000000000 & -0.000000000000000 & 0.000000000000000 & -0.683270656201011 \end{bmatrix}.$$

Assim, a matriz  $Q$  é

$$Q \cong \begin{bmatrix} 0.179605302026775 & 0.359210604053550 & 0.898026510133874 & 0.179605302026775 \\ 0.377964473009227 & 0.377964473009227 & -0.377964473009227 & 0.755928946018455 \\ 0.806556388364954 & 0.175226308015390 & -0.121112301128284 & -0.551447498754314 \\ 0.417554289900618 & -0.835108579801236 & 0.189797404500281 & 0.303675847200449 \end{bmatrix}$$

obtendo-se a fatorização  $QR$  da matriz  $A$ .

Finalmente, descrevemos o método que permite obter a fatorização QR com recurso a matrizes de Givens (Definição 2.18). O processo é análogo ao relatado usando matrizes de Householder e explicamo-lo de seguida.

Relembramos que dado um vetor  $x = (x_1, \dots, x_n)$  as coordenadas do vetor  $y = Gx$ , com  $G \equiv G(p, q, \theta)$  uma matriz de Givens, são dadas por

$$y_j = \begin{cases} x_j & , j \neq p, q \\ \cos(\theta)x_p - \sin(\theta)x_q & , j = p \\ \sin(\theta)x_p + \cos(\theta)x_q & , j = q \end{cases} . \quad (6.21)$$

Se pretendermos anular a componente  $y_p$  devemos ter  $\cos(\theta)x_p - \sin(\theta)x_q = 0$ , ou seja,  $\tan(\theta) = \frac{x_p}{x_q}$ . Assim, tomamos

$$\theta = \arctan\left(\frac{x_p}{x_q}\right),$$

desde que  $x_q \neq 0$ . Caso  $x_q = 0$  tomamos  $\theta = \frac{\pi}{2}$ .

Da mesma forma, se quisermos anular a componente  $y_q$  tomamos

$$\theta = \arctan\left(-\frac{x_q}{x_p}\right) = -\arctan\left(\frac{x_q}{x_p}\right),$$

desde que  $x_p \neq 0$ . Caso  $x_p = 0$  tomamos  $\theta = \frac{\pi}{2}$ .

Como nem sempre podemos ter a garantia de que as componentes  $x_p$  ou  $x_q$ , do vetor  $x$ , são não nulas podemos analisar a situação de uma outra forma. Se tomarmos

$$\cos(\theta) = \frac{x_q}{\sqrt{x_p^2 + x_q^2}}$$

e

$$\sin(\theta) = \frac{x_p}{\sqrt{x_p^2 + x_q^2}}$$

as componentes do vetor  $y$  serão dadas por

$$\begin{cases} y_p & = 0 \\ y_q & = \sqrt{x_p^2 + x_q^2} \\ y_j & = x_j \end{cases} , \text{para } j \neq p, q .$$

Escolhendo  $\theta$  desta forma poderemos anular a componente da posição  $p$ .

Se, por outro lado, tomarmos

$$\cos(\theta) = \frac{x_p}{\sqrt{x_p^2 + x_q^2}}$$

e

$$\sin(\theta) = -\frac{x_q}{\sqrt{x_p^2 + x_q^2}}$$

as componentes do vetor  $y$  são

$$\begin{cases} y_p & = \sqrt{x_p^2 + x_q^2} \\ y_q & = 0 \\ y_j & = x_j \end{cases} , \text{para } j \neq p, q$$

e assim anulamos a componente da posição  $q$ .

**Exemplo 6.7** Tomemos o vetor  $x = [1 \ 1 \ 1 \ 1]^T$ . Vamos anular a sua última componente. Para tal, vamos tomar

$$G = G(3, 4, \theta) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & \cos(\theta) & -\sin(\theta) \\ 0 & 0 & \sin(\theta) & \cos(\theta) \end{bmatrix}$$

onde  $\cos(\theta) = \frac{1}{\sqrt{2}}$  e  $\sin(\theta) = -\frac{1}{\sqrt{2}}$ . Desta forma,

$$G = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ 0 & 0 & -\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{bmatrix}.$$

Assim,

$$y = Gx = [1 \ 1 \ \sqrt{2} \ 0]^T.$$

Como facilmente se constata que a aplicação sucessiva de transformações de Givens, devidamente escolhidas, permitem reduzir uma matriz  $A$  a uma matriz triangular, e tendo em conta que as matrizes de Givens são ortogonais, é possível obter a decomposição QR de  $A$ , tal como no método descrito anteriormente, quando usámos as matrizes de Householder. Mais precisamente, este processo produz uma sucessão de matrizes  $A^{(k)}$  nas quais iremos atuar sobre as colunas com o objetivo de provocar o anulamento dos elementos abaixo da diagonal principal.

Vamos aqui explicar como proceder para a primeira coluna. Começamos por anular o elemento  $a_{1n}$  usando, para tal, o elemento  $a_{1,n-1}$  e construindo a matriz de rotação  $G(n-1, n, \theta_1) \equiv G_1$ . De seguida, obtemos a matriz  $A^{(1)} = G_1 A$ , cujo elemento  $a_{1n}^{(1)} = 0$ .

Da matriz  $A^{(1)}$  pretendemos anular o elemento  $a_{1,n-1}^{(1)}$ , usando o elemento  $a_{1,n-2}^{(1)}$  e construindo a matriz de rotação  $G(n-2, n-1, \theta_2) \equiv G_2$ . Obtemos a matriz  $A^{(2)} = G_2 A^{(1)}$  cujos elementos  $a_{1n}^{(2)} = a_{1,n-1}^{(2)} = 0$ , pois este processo preserva os zeros obtidos anteriormente já que a operação  $G_2 A^{(1)}$  irá apenas alterar as linhas  $(n-2)$  e  $(n-1)$  da matriz  $A^{(1)}$ .

Até à linha 2 procedemos da mesma forma e obtemos a matriz  $A^{(n-1)} = G_{n-1} A^{(n-2)}$ .

Este processo repete-se nas colunas seguintes até à coluna  $(n-1)$  onde se irá anular apenas o último elemento da matriz resultante. No final, teremos

$$\begin{array}{cccc} A^{(\frac{n(n-1)}{2})} = & G_{\frac{n(n-1)}{2}} & \cdots & (G_{2n-3} \cdots G_n) \quad (G_{n-1} \cdots G_2 G_1) A \\ & \downarrow & \cdots & \downarrow \quad \downarrow \\ & \text{Coluna } (n-1) & \cdots & \text{Coluna 2} \quad \text{Coluna 1} \end{array}$$

onde, se fizermos

$$\begin{aligned} Q_1 &= G_{n-1} \cdots G_2 G_1 \\ Q_2 &= G_n \cdots G_{2n-3} \\ &\vdots \\ Q_{n-1} &= G_{\frac{n(n-1)}{2}}, \end{aligned}$$

obtemos

$$\begin{aligned} R \equiv A^{(\frac{n(n-1)}{2})} &= Q_{n-1}^T Q_{n-2}^T \cdots Q_1^T A \\ &= (Q_1 \cdots Q_{n-2} Q_{n-1})^T A \\ &= Q^T A \end{aligned} \tag{6.22}$$

e a matriz  $R$  está na forma triangular superior. Como a matriz  $Q$  é produto de matrizes ortogonais, pela Propriedade 2.5, também ela é ortogonal e então  $Q^{-1} = Q^T$ , podendo reescrever-se (6.22) da seguinte forma

$$A = QR.$$

Obtivemos a fatorização QR da matriz  $A$ , usando matrizes de rotação de Givens.

**Exemplo 6.8** Para que o exemplo não fique muito longo iremos apresentar a fatorização QR, com recurso a rotações de Givens, para uma matriz  $3 \times 3$ . Consideremos a matriz

$$B = \begin{bmatrix} 1 & 2 & 3 \\ 3 & 5 & 1 \\ 2 & 2 & 1 \end{bmatrix}$$

• 1<sup>a</sup> Coluna

$$\begin{aligned} G_1 &\cong \begin{bmatrix} 1.000000000000000 & 0 & 0 \\ 0 & 0.832050294337844 & 0.554700196225229 \\ 0 & -0.554700196225229 & 0.832050294337844 \end{bmatrix}, \\ B^{(1)} &\cong \begin{bmatrix} 1.000000000000000 & 2.000000000000000 & 3.000000000000000 \\ 3.605551275463990 & 5.269651864139677 & 1.386750490563073 \\ 0 & -1.109400392450459 & 0.277350098112615 \end{bmatrix}, \\ G_2 &\cong \begin{bmatrix} 0.267261241912424 & 0.963624111659432 & 0 \\ -0.963624111659432 & 0.267261241912424 & 0 \\ 0 & 0 & 1.000000000000000 \end{bmatrix}, \\ B^{(2)} &\cong \begin{bmatrix} 3.741657386773941 & 5.612486080160912 & 2.138089935299395 \\ -0.000000000000000 & -0.518874521662771 & -2.520247676647744 \\ 0 & -1.109400392450459 & 0.277350098112615 \end{bmatrix}; \end{aligned}$$



- 2ª Coluna

$$G_3 \cong \begin{bmatrix} 1.000000000000000 & 0 & 0 \\ 0 & -0.423659272868162 & -0.905821627315677 \\ 0 & 0.905821627315677 & -0.423659272868162 \end{bmatrix},$$

$$R \equiv B^{(3)} \cong \begin{bmatrix} 3.741657386773941 & 5.612486080160912 & 2.138089935299395 \\ 0.000000000000000 & 1.224744871391589 & 0.816496580927726 \\ -0.000000000000000 & -0.000000000000000 & -2.400396792595917 \end{bmatrix}.$$

Obtemos a matriz  $R$  triangular superior e para a matriz  $Q$  temos

$$Q = G_3 G_2 G_1 \cong \begin{bmatrix} 0.267261241912424 & 0.801783725737273 & 0.534522483824849 \\ 0.408248290463863 & 0.408248290463863 & -0.816496580927726 \\ -0.872871560943970 & 0.436435780471985 & -0.218217890235992 \end{bmatrix},$$

obtendo-se a fatorização QR da matriz  $B$ .

### 6.2.2 O Método QR Básico

Seja  $A \in \mathbb{R}^{n \times n}$  uma matriz invertível. Começamos por tomar uma matriz ortogonal  $Q^{(0)} \in \mathbb{R}^{n \times n}$  e fazemos

$$T^{(0)} = (Q^{(0)})^T A Q^{(0)},$$

onde, por exemplo, podemos tomar  $Q^{(0)} = I_n$ .

O passo a seguir é o de obter a fatorização QR da matriz  $T^{(0)}$ ,

$$T^{(0)} = Q^{(1)} R^{(1)},$$

onde  $Q^{(1)}$  é uma matriz ortogonal e  $R^{(1)}$  é uma matriz triangular superior.

De seguida, obtemos a matriz  $T^{(1)}$  da seguinte forma

$$T^{(1)} = R^{(1)} Q^{(1)}.$$

Para esta nova matriz encontramos a sua decomposição QR, isto é, determinamos as matrizes  $Q^{(2)}$  ortogonal e  $R^{(2)}$  triangular superior tais que

$$T^{(1)} = Q^{(2)} R^{(2)}$$

e calculamos a matriz  $T^{(2)}$  como indicado a seguir

$$T^{(2)} = R^{(2)} Q^{(2)},$$

e assim sucessivamente, obtendo-se o processo iterativo que indicamos a seguir.

Para  $k = 1, 2, \dots$

1. determinar as matrizes  $Q^{(k)}$  e  $R^{(k)}$  através da decomposição QR de  $T^{(k-1)}$ :

$$T^{(k-1)} = Q^{(k)} R^{(k)};$$

2. fazer

$$T^{(k)} = R^{(k)}Q^{(k)}.$$

De notar que toda a matriz  $T^{(k)}$  é ortogonalmente semelhante à matriz inicial  $A$ , uma vez que

$$\begin{aligned} T^{(k)} &= R^{(k)}Q^{(k)} \\ &= (Q^{(k)})^T (Q^{(k)}R^{(k)}) Q^{(k)}, \text{ pois } Q^{(k)} \text{ é ortogonal} \\ &= (Q^{(k)})^T T^{(k-1)}Q^{(k)}, \end{aligned} \quad (6.23)$$

e, usando indução em  $k$ , facilmente mostramos que

$$T^{(k)} = (P^{(k)})^T A P^{(k)} \quad (6.24)$$

com  $P^{(k)} = Q^{(0)}Q^{(1)} \dots Q^{(k-1)}Q^{(k)}$ ,  $k \geq 0$ , que pela Propriedade 2.5, continua a ser uma matriz ortogonal.

A decomposição QR, pedida no método, pode ser obtida por qualquer um dos processos descritos na Secção 6.2.1 (Processo de Ortonormalização de Gram-Schmidt, Matrizes de Reflexão de Householder ou Matrizes de Rotação de Givens).

**Exemplo 6.9** *Voltemos à matriz (4.11). Vamos aplicar o Método QR Básico para obtermos uma aproximação dos seus valores próprios. Para a resolução deste exemplo, utilizámos o comando `qr(A)` do MATLAB®, que devolve a matriz  $Q$  ortogonal e a matriz  $R$  triangular superior da fatorização QR da matriz  $A$ . Ao fim de 20 iterações, obtivemos os seguintes resultados*

$$\begin{aligned} \lambda_1 &\cong 11.840474193588957 \\ \lambda_2 &\cong 4.018097046416818 \\ \lambda_3 &\cong -2.563382668194998 \\ \lambda_4 &\cong -0.295188571810783. \end{aligned}$$

*Podemos verificar que estes valores estão relativamente próximos aos obtidos em (3.14). É claro que se aplicarmos o método, por exemplo, em 50 iterações a aproximação melhora.*

**Exemplo 6.10** *Retomemos a matriz (6.18).*

*Aplicado o Método QR Básico obtemos, após 50000 iterações, a seguinte matriz*

$$T^{(50000)} \cong \begin{bmatrix} -0.260233606969402 & -3.380536047837450 & 0.582756133204716 & -0.954262519769544 \\ 1.886664022329994 & -0.318911419042252 & -4.067535405210567 & 0.901840258797449 \\ 0 & 0.000000000000000 & 2.381178355535637 & 0.625500041584830 \\ 0 & 0 & -1.530473434220873 & 2.197966670476156 \end{bmatrix}.$$

Verificamos que não se trata de uma matriz triangular superior mas sim de uma matriz triangular por blocos  $2 \times 2$ . Desses blocos,

$$T_1 = \begin{bmatrix} -0.260233606969402 & -3.380536047837450 \\ 1.886664022329994 & -0.318911419042252 \end{bmatrix}$$

e

$$T_2 = \begin{bmatrix} 2.381178355535637 & 0.625500041584830 \\ -1.530473434220873 & 2.197966670476156 \end{bmatrix},$$

podemos obter os valores próprios de  $B$ ,

$$\begin{aligned} \lambda_1 &\cong -0.289572513005827 + 2.525287105704444 i \\ \lambda_2 &\cong -0.289572513005827 - 2.525287105704444 i \\ \lambda_3 &\cong 2.289572513005897 + 0.974125026043397 i \\ \lambda_4 &\cong 2.289572513005897 - 0.974125026043397 i. \end{aligned}$$

A razão da ocorrência deste tipo de matriz por blocos será explicada mais adiante, na Secção 6.2.4.

### 6.2.3 O Método Hessenberg-QR

Uma outra forma de iniciarmos o processo iterativo do Método QR é tomando  $T^{(0)} = H^{(0)}$ , com  $H^{(0)}$  uma matriz ortogonalmente semelhante a  $A$  na forma superior de Hessenberg (Definição 2.15). A matriz  $H^{(0)}$  é obtida a partir de  $A$  por reflexões de Householder e, como no caso anterior, a decomposição QR pedida pelo método iterativo QR pode ser obtida por qualquer um dos métodos anteriores. Este método iterativo é também conhecido como Método Iterativo Hessenberg-QR.

Mostramos aqui como é que se obtém por semelhança uma matriz de Hessenberg a partir de uma matriz  $A$ , dada. Este processo de redução é conhecido por transformação ou método de Householder, visto que utiliza matrizes de reflexão de Householder.

Tomemos uma qualquer matriz  $A \in \mathbb{R}^{n \times n}$ . O método consiste em aplicar, sucessivamente, matrizes de Householder de forma a anular, convenientemente, as entradas da matriz até reduzi-la à forma superior de Hessenberg. Da mesma forma vamos aplicar um método semelhante ao descrito, da página 82 à página 84, a cada coluna da matriz  $A$ ,

$$A = [a_{ij}] = \begin{bmatrix} A_1 & A_2 & \cdots & A_{n-1} & A_n \end{bmatrix},$$

com  $A_i$ ,  $i = 1, \dots, n$ , os vetores coluna que formam a matriz  $A$ .

Começamos por encontrar a matriz de reflexão  $P_1$  de forma a anular as  $(n-2)$  últimas componentes do vetor coluna  $A_1$ , obtendo uma nova matriz  $A^{(1)} = \begin{bmatrix} a_{ij}^{(1)} \end{bmatrix} = P_1^T A P_1$ , com uma coluna reduzida. Isto é,

$$\begin{aligned} A^{(1)} &= P_1^T A P_1 \\ &= P_1 A P_1, \end{aligned}$$

onde a matriz  $P_1$  é ortogonal e da forma

$$P_1 = \left[ \begin{array}{c|c} 1 & 0 \\ \hline 0 & \tilde{P}_1 \end{array} \right],$$

sendo  $\tilde{P}_1$  a matriz de Householder que anula as últimas  $(n-2)$  componentes de

$$x_1 = [a_{21} \ a_{31} \ \cdots \ a_{n1}]^T.$$

Isto é,  $\tilde{P}_1$  é tal que

$$\tilde{P}_1 x_1 = [||x_1||_2 \ 0 \ \cdots \ 0]^T.$$

Assim,

$$\begin{aligned} A^{(1)} &= \left[ \begin{array}{c|c} 1 & 0 \\ \hline 0 & \tilde{P}_1 \end{array} \right] \left[ \begin{array}{c|c} a_{11} & *1 \\ \hline a_{21} & *2 \\ \vdots & \\ a_{n1} & \end{array} \right] \left[ \begin{array}{c|c} 1 & 0 \\ \hline 0 & \tilde{P}_1 \end{array} \right] \\ &= \left[ \begin{array}{c|c} a_{11} & *1 \\ \hline \tilde{P}_1 x_1 & \tilde{P}_1 *2 \end{array} \right] \left[ \begin{array}{c|c} 1 & 0 \\ \hline 0 & \tilde{P}_1 \end{array} \right] \\ &= \left[ \begin{array}{c|c} a_{11} & *1 \tilde{P}_1 \\ \hline ||x||_2 & \\ 0 & \\ \vdots & \tilde{P}_1 *2 \tilde{P}_1 \\ 0 & \end{array} \right], \end{aligned}$$

ou seja,  $A^{(1)}$  será da forma

$$\left[ \begin{array}{c|c} * & \tilde{*}_1 \\ \hline * & \\ 0 & \\ \vdots & \tilde{*}_2 \\ 0 & \end{array} \right].$$

De seguida, encontramos a matriz de Householder,  $P_2$ , que permite anular as  $(n-3)$  últimas componentes da segunda coluna de  $A^{(1)}$ , obtendo a matriz  $A^{(2)} = [a_{ij}^{(2)}] = P_2^T A^{(1)} P_2$ , com duas colunas reduzidas, visto que a aplicação da matriz  $P_2$  à matriz  $A^{(1)}$  preserva os zeros obtidos anteriormente. Concretizando, vamos encontrar a matriz de Householder

$$P_2 = \left[ \begin{array}{c|c} I_2 & 0 \\ \hline 0 & \tilde{P}_2 \end{array} \right]$$

onde  $\tilde{P}_2$  é a matriz de Householder que anula as últimas  $(n - 3)$  componentes de

$$x_2 = \begin{bmatrix} a_{32}^{(1)} & a_{42}^{(1)} & \cdots & a_{n2}^{(1)} \end{bmatrix}^T.$$

Isto é,  $\tilde{P}_2$  é tal que

$$\tilde{P}_2 x_2 = \begin{bmatrix} \|x_2\|_2 & 0 & \cdots & 0 \end{bmatrix}^T.$$

Assim, e porque  $P_2$  é ortogonal,

$$\begin{aligned} A^{(2)} &= P_2^T A^{(1)} P_2 \\ &= P_2 A^{(1)} P_2 \\ &= \begin{bmatrix} I_2 & 0 \\ 0 & \tilde{P}_2 \end{bmatrix} \begin{bmatrix} a_{11}^{(1)} & a_{12}^{(1)} & *1 \\ a_{21}^{(1)} & a_{22}^{(1)} & \\ 0 & a_{32}^{(1)} & *2 \\ \vdots & \vdots & \\ 0 & a_{n2}^{(1)} & \end{bmatrix} \begin{bmatrix} I_2 & 0 \\ 0 & \tilde{P}_2 \end{bmatrix} \\ &= \begin{bmatrix} a_{11}^{(1)} & a_{12}^{(1)} & *1 \\ a_{21}^{(1)} & a_{22}^{(1)} & \\ 0 & \tilde{P}_2 x_2 & \tilde{P}_2 *2 \\ \vdots & & \\ 0 & & \end{bmatrix} \begin{bmatrix} I_2 & 0 \\ 0 & \tilde{P}_2 \end{bmatrix} \\ &= \begin{bmatrix} a_{11}^{(1)} & a_{12}^{(1)} & *1 \tilde{P}_2 \\ a_{21}^{(1)} & a_{22}^{(1)} & \\ 0 & \|x_2\|_2 & \\ 0 & 0 & \\ \vdots & \vdots & \tilde{P}_2 *2 \tilde{P}_2 \\ 0 & 0 & \end{bmatrix}, \end{aligned}$$

ou seja,  $A^{(2)}$  será da forma

$$\begin{bmatrix} * & * & \tilde{*}_1 \\ * & * & \\ 0 & * & \\ 0 & 0 & \\ \vdots & \vdots & \tilde{*}_2 \\ 0 & 0 & \end{bmatrix}.$$

O processo continua até à  $(n - 2)$ -ésima coluna da matriz  $A^{(n-3)}$  de forma a anular a sua última componente. Assim, em  $(n - 2)$  passos temos a matriz  $A$  reduzida à forma superior de Hessenberg,

$$H \equiv A^{(n-2)} = P_{n-2}^T A^{(n-3)} P_{n-2}.$$

Neste último passo, encontramos a matriz de reflexão  $P_{n-2}$  que anula a última componente da  $(n-3)$ -ésima coluna da matriz  $A^{(n-3)} = [a_{ij}^{(n-3)}]$ , ou seja, construímos a matriz de Householder

$$P_{n-2} = \left[ \begin{array}{c|c} I_{n-2} & 0 \\ \hline 0 & \tilde{P}_{n-2} \end{array} \right],$$

onde  $\tilde{P}_{n-2}$  é a matriz de reflexão que anula a última componente de

$$x_{n-2} = \begin{bmatrix} a_{n-1,n-2}^{(n-3)} & a_{n,n-2}^{(n-3)} \end{bmatrix}^T.$$

Isto é,  $\tilde{P}_{n-2}$  é tal que

$$\tilde{P}_{n-2}x_{n-2} = \begin{bmatrix} \|x_{n-2}\|_2 & 0 \end{bmatrix}^T.$$

Como  $P_{n-2}$  é ortogonal, temos

$$\begin{aligned} A^{(n-2)} &= P_{n-2}^T A^{(n-3)} P_{n-2} \\ &= P_{n-2} A^{(n-3)} P_{n-2} \\ &= \left[ \begin{array}{c|c} I_{n-2} & 0 \\ \hline 0 & \tilde{P}_{n-2} \end{array} \right] \left[ \begin{array}{cccc|c} A_{n-2,n-3}^{(n-3)} & & & & *1 \\ 0 & \cdots & 0 & a_{n-1,n-3}^{(n-3)} & *2 \\ 0 & \cdots & 0 & a_{n,n-3}^{(n-3)} & \end{array} \right] \left[ \begin{array}{c|c} I_{n-2} & 0 \\ \hline 0 & \tilde{P}_{n-2} \end{array} \right] \\ &= \left[ \begin{array}{cccc|c} A_{n-2,n-3}^{(n-3)} & & & & *1 \\ 0 & \cdots & 0 & \tilde{P}_{n-2}x_{n-2} & \tilde{P}_{n-2}*2 \\ 0 & \cdots & 0 & & \end{array} \right] \left[ \begin{array}{c|c} I_{n-2} & 0 \\ \hline 0 & \tilde{P}_{n-2} \end{array} \right] \\ &= \left[ \begin{array}{cccc|c} A_{n-2,n-3}^{(n-3)} & & & & *1\tilde{P}_{n-2} \\ 0 & \cdots & 0 & \|x_{n-2}\|_2 & \tilde{P}_{n-2}*2\tilde{P}_{n-2} \\ 0 & \cdots & 0 & 0 & \end{array} \right], \end{aligned}$$

ou seja,  $H \equiv A^{(n-2)}$  está na forma superior de Hessenberg, como pretendíamos,

$$\begin{bmatrix} * & * & \cdots & \cdots & \cdots & * \\ * & * & \cdots & \cdots & \cdots & * \\ 0 & * & \ddots & & & \vdots \\ 0 & 0 & \ddots & \ddots & & \vdots \\ \vdots & \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 & * & * \end{bmatrix}.$$

Notamos que, para qualquer  $k \geq 1$ , a matriz  $A^{(k)}$  é ortogonalmente semelhante a  $A$ . De facto,

$$\begin{aligned} A^{(k)} &= P_k^T A^{(k-1)} P_k \\ &= P_k^T (P_{k-1}^T A^{(k-1)} P_{k-1}) P_k \\ &\vdots \\ &= (P_1 \cdots P_k)^T A (P_1 \cdots P_k) \\ &= O^T A O, \end{aligned}$$

onde  $O = P_1 \cdots P_k$  é uma matriz ortogonal pois cada  $P_i, i = 1, \dots, k$ , é ortogonal (Propriedade 2.5).

**Exemplo 6.11** Neste exemplo, vamos aplicar o Método QR à matriz (4.11), com esta reduzida à forma superior de Hessenberg. Começamos pela transformação de semelhança da matriz  $A$  numa matriz de Hessenberg  $H$ .

Para a primeira redução, a matriz de Householder é

$$P_1 \cong \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0.365148371670111 & 0.912870929175277 & 0.182574185835055 \\ 0 & 0.912870929175277 & -0.312642665067414 & -0.262528533013483 \\ 0 & 0.182574185835055 & -0.262528533013483 & 0.947494293397303 \end{bmatrix}$$

e a matriz  $A^{(1)} = P_1^T A P_1$  é

$$A^{(1)} \cong \begin{bmatrix} 1.000000000000000 & 5.477225575051662 & -0.000000000000000 & 0 \\ 5.477225575051662 & 8.100000000000001 & 2.173633014644720 & 1.931834926776398 \\ -0.000000000000000 & 2.173633014644720 & -0.292579853359375 & 0.952191648062500 \\ 0 & 1.931834926776398 & 0.952191648062499 & 4.192579853359375 \end{bmatrix}.$$

Para a segunda, iteração a matriz de Householder é

$$P_2 \cong \begin{bmatrix} 1.000000000000000 & 0 & 0 & 0 \\ 0 & 1.000000000000000 & 0 & 0 \\ 0 & 0 & 0.747457692087519 & 0.664309414760321 \\ 0 & 0 & 0.664309414760321 & -0.747457692087518 \end{bmatrix}$$

e a matriz  $H \equiv A^{(2)} = P_2^T A P_2$  é

$$H \cong \begin{bmatrix} 1.000000000000000 & 5.477225575051662 & -0.000000000000000 & -0.000000000000000 \\ 5.477225575051662 & 8.100000000000001 & 2.908034846191955 & -0.000000000000001 \\ -0.000000000000000 & 2.908034846191955 & 2.632361056365784 & -2.338849444083287 \\ -0.000000000000000 & -0.000000000000000 & -2.338849444083287 & 1.267638943634214 \end{bmatrix}.$$

Aplicamos, agora, o método QR à matriz  $H$  tal como foi descrito na Subsecção 6.2.2 e obtemos

$$\begin{aligned} \lambda_1 &\cong 11.840474193588967 \\ \lambda_2 &\cong 4.018097046416803 \\ \lambda_3 &\cong -2.563382668194985 \\ \lambda_4 &\cong -0.295188571810783. \end{aligned}$$

### 6.2.4 A convergência do Método QR

Podemos encontrar uma demonstração geral da convergência do Método Iterativo QR em [29, pp. 517–523]. No entanto, essa demonstração é longa e envolve temas que não são contemplados neste trabalho como, por exemplo, técnicas de pivotagem. O autor prova que se a matriz  $A$  é diagonalizável então a sequência de matrizes  $T^{(k)}$ , do método QR, é sempre convergente para uma matriz triangular por blocos em que cada bloco da diagonal tem por valores próprios os valores próprios de  $A$  iguais entre si, em módulo. Isto é, se agruparmos os valores próprios de  $A$ , formando conjuntos de valores de igual módulo, por exemplo,

$$\begin{array}{ll} \lambda_1, \dots, \lambda_{s_1} & \text{com módulo } |\lambda_1| \\ \lambda_2, \dots, \lambda_{s_2} & \text{com módulo } |\lambda_2| \\ \vdots & \\ \lambda_m, \dots, \lambda_{s_m} & \text{com módulo } |\lambda_m| \end{array}$$

e  $|\lambda_i| \neq |\lambda_j|$ , para  $i \neq j$  e  $i, j = 1, \dots, m$ , contando as multiplicidades, o método QR converge para uma matriz da forma

$$\begin{bmatrix} T_1 & * & \cdots & \cdots & * \\ 0 & T_2 & \ddots & & \vdots \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \vdots & & \ddots & T_{m-1} & * \\ 0 & \cdots & \cdots & 0 & T_m \end{bmatrix},$$

onde  $T_i \in \mathbb{R}^{s_i \times s_i}$ ,  $i = 1, \dots, m$ , e  $\sigma(T_i) = \{\lambda_i, \dots, \lambda_{s_i}\}$ . A parte mais simples da demonstração é tratada em quase toda a bibliografia que trata o tema. Essa será a prova que apresentamos de seguida, seguindo o autor [22, pp. 117–118]. Para provarmos a convergência do Método QR vamos fixar-nos no comportamento da sucessão de matrizes  $T^{(k)}$ . Recordamos que a iteração QR (básica), com  $Q^{(0)} = I_n$ , é a seguinte

$$\begin{aligned} T^{(0)} &= A \\ &= Q^{(1)}R^{(1)} \\ \\ T^{(1)} &= R^{(1)}Q^{(1)} \\ &= Q^{(2)}R^{(2)} \\ &\vdots \\ T^{(k-1)} &= R^{(k-1)}Q^{(k-1)} \\ &= Q^{(k)}R^{(k)}, \quad k = 1, 2, 3, \dots \end{aligned}$$

O lema seguinte permite identificar a decomposição QR de qualquer potência de  $A$ , usando as matrizes  $Q^{(k)}$  e  $R^{(k)}$  que se vão obtendo neste processo iterativo.



**Lema 6.1** *Sejam  $A \in \mathbb{R}^{n \times n}$  uma matriz invertível e*

$$T^{(k-1)} = Q^{(k)} R^{(k)}, \quad k \geq 1,$$

*a sucessão de matrizes dadas pela iteração QR. Defina-se*

$$P^{(k)} = Q^{(1)} \cdots Q^{(k)}, \quad k = 1, 2, 3, \dots$$

*e*

$$U^{(k)} = R^{(k)} \cdots R^{(1)}, \quad k = 1, 2, 3, \dots$$

*Então  $P^{(k)} U^{(k)}$  é a fatorização QR da matriz  $A^k$ , isto é,  $A^k = P^{(k)} U^{(k)}$ , para  $k = 1, 2, 3, \dots$*

**Prova:** Sabemos, por (6.24), que

$$T^{(k-1)} = (P^{(k-1)})^T A P^{(k-1)}, \quad k = 1, 2, \dots,$$

convencionando que  $P^{(0)} = I_n$  e também que  $U^{(0)} = I_n$ . Logo, como  $P^{(k-1)}$  é ortogonal

$$P^{(k-1)} T^{(k-1)} = A P^{(k-1)}. \quad (6.25)$$

Por outro lado,

$$\begin{aligned} P^{(k)} U^{(k)} &= P^{(k-1)} (Q^{(k)} R^{(k)}) U^{(k-1)} \\ &= (P^{(k-1)} T^{(k-1)}) U^{(k-1)}. \end{aligned}$$

Assim, por (6.25),

$$P^{(k)} U^{(k)} = A P^{(k-1)} U^{(k-1)}. \quad (6.26)$$

Usando (6.26), obtemos, fazendo  $k = 1$ ,

$$P^{(1)} U^{(1)} = A,$$

ou seja,

$$Q^{(1)} R^{(1)} = A,$$

que é a decomposição QR de  $A$ , como deveria ser. Fazendo  $k = 2$ ,

$$\begin{aligned} P^{(2)} U^{(2)} &= A P^{(1)} U^{(1)} \\ &= A^2, \end{aligned}$$

e assim sucessivamente. Isto é, usando indução sobre  $k$ , facilmente se conclui que

$$P^{(k)} U^{(k)} = A^k,$$

para  $k = 1, 2, \dots$  □

De seguida, enunciamos e provamos o teorema que nos dá as condições para a convergência do Método QR.

**Teorema 6.4** *Seja  $A \in \mathbb{R}^{n \times n}$  uma matriz invertível cujos valores próprios são distintos, em módulo, ou seja,*

$$|\lambda_1| > |\lambda_2| > \cdots > |\lambda_n|. \quad (6.27)$$

*Se existir a matriz  $Y$  invertível que admita fatorização  $LU$ , tal que*

$$A = Y^{-1} \text{diag}(\lambda_1, \dots, \lambda_n) Y$$

*então a sucessão de matrizes  $T^{(k)}$ ,  $k = 0, 1, \dots$ , tende para uma matriz triangular superior onde os valores próprios de  $A$  são os elementos da diagonal principal.*

**Prova:** Por hipótese, seja  $Y = LU$  a fatorização LU da matriz  $Y$  e seja  $Y^{-1} = QR$  a fatorização QR da matriz  $Y^{-1}$ .

Como

$$A^k = Y^{-1} (\text{diag}(\lambda_1, \dots, \lambda_n))^k Y,$$

ou fazendo  $D = \text{diag}(\lambda_1, \dots, \lambda_n)$ ,

$$A^k = Y^{-1} D^k Y,$$

pelo Lema 6.1

$$\begin{aligned} P^{(k)} U^{(k)} &= Y^{-1} D^k Y \\ &= (QR) D^k (LU) \\ &= (QR) (D^k L D^{-k}) (D^k U). \end{aligned} \quad (6.28)$$

A matriz  $D^k L D^{-k}$  é triangular inferior com elementos unitários na diagonal principal e os elementos abaixo da diagonal principal tendem para zero. De facto, como

$$\begin{aligned} D^k &= \text{diag}(\lambda_1^k, \dots, \lambda_n^k), \\ D^{-k} &= \text{diag}(\lambda_1^{-k}, \dots, \lambda_n^{-k}) \end{aligned}$$

e

$$L = \begin{bmatrix} 1 & 0 & \cdots & \cdots & 0 \\ \ell_{21} & 1 & \ddots & & \vdots \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \vdots & & \ddots & 1 & 0 \\ \ell_{n1} & \cdots & \cdots & \ell_{n,n-1} & 1 \end{bmatrix}$$

vamos ter

$$D^k L D^{-k} = [\tilde{\ell}_{ij}^{(k)}], i, j = 1, \dots, n,$$

onde

$$\tilde{\ell}_{ij}^{(k)} = \begin{cases} 0 & \text{se } j > i \\ 1 & \text{se } i = j \\ \frac{\lambda_i^k}{\lambda_j^k} \ell_{ij} & \text{se } i > j \end{cases}.$$

Assim, por (6.27), temos  $\left| \frac{\lambda_i}{\lambda_j} \right| < 1$ , com  $i > j$ , e então  $\tilde{\ell}_{ij}^{(k)} \rightarrow 0$ . Logo,

$$D^k L D^{-k} = I_n + E^{(k)},$$

com  $\lim_{k \rightarrow \infty} E^{(k)} = 0_n$ .

Assim, podemos reescrever (6.28) da seguinte forma

$$\begin{aligned} P^{(k)} U^{(k)} &= Q R (I_n + E^{(k)}) D^k U \\ &= Q (R I_n + R E^{(k)}) R^{-1} R D^k U \\ &= Q (I_n + R E^{(k)} R^{-1}) R D^k U \\ &= Q (I_n + F^{(k)}) R D^k U, \end{aligned}$$

com  $F^{(k)} = R E^{(k)} R^{-1}$ . Como  $E^{(k)} \rightarrow 0_n$  então

$$F^{(k)} = R E^{(k)} R^{-1} \rightarrow 0_n. \quad (6.29)$$

Seja  $O^{(k)} V^{(k)} = I_n + F^{(k)}$  a fatorização QR da matriz  $(I_n + F^{(k)})$ . Então

$$\lim_{k \rightarrow \infty} O^{(k)} V^{(k)} = \lim_{k \rightarrow \infty} O^{(k)} \lim_{k \rightarrow \infty} V^{(k)} = O V,$$

e as matrizes  $O$  e  $V$  existem por continuidade e são, respetivamente, ortogonal e triangular superior. Por (6.29),  $O V = I_n$  e como a decomposição QR é única e  $I_n = I_n I_n$ , com  $I_n$  simultaneamente ortogonal e triangular superior, temos  $O = I_n$  e  $V = I_n$ .

Assim,

$$\begin{aligned} P^{(k)} U^{(k)} &= Q (I_n + F^{(k)}) R D^k U \\ &= (Q O^{(k)}) (V^{(k)} R D^k U), \end{aligned} \quad (6.30)$$

onde  $Q O^{(k)}$  é uma matriz ortogonal e  $V^{(k)} R D^k U$  é triangular.

Vamos considerar  $|D| = \text{diag}(|\lambda_1|, \dots, |\lambda_n|)$  e definir

$$D_1 := |D|^{-1} D = \text{diag} \left( \frac{\lambda_1}{|\lambda_1|}, \dots, \frac{\lambda_n}{|\lambda_n|} \right),$$

$$D_2 := \text{diag} \left( \frac{u_{11}}{|u_{11}|}, \dots, \frac{u_{nn}}{|u_{nn}|} \right),$$

para  $U = [u_{ij}]$ ,  $i, j = 1, \dots, n$ , e ainda

$$\begin{aligned} \tilde{U} &:= (D_2)^{-1} U \\ &= \text{diag} \left( \frac{|u_{11}|}{u_{11}}, \dots, \frac{|u_{nn}|}{u_{nn}} \right) U. \end{aligned}$$

Salientamos que as matrizes  $D_1$  e  $D_2$  são ortogonais e que, ao efetuarmos a operação  $(D_2)^{-1}U$ , os elementos da diagonal principal da matriz  $\tilde{U}$  são  $|u_{ii}|, i = 1, \dots, n$ , ou seja, os elementos da diagonal principal de  $\tilde{U}$  são positivos.

De (6.30) e da unicidade da decomposição QR, temos que

$$P^{(k)} = QO^{(k)}D_1^kD_2$$

e

$$U^{(k)} = (D_1^kD_2)^{-1}V^{(k)}RD^kU$$

definem a decomposição QR de  $A^k$ . Mais, recordando que matrizes diagonais permutam entre si,

$$\begin{aligned} U^{(k)} &= (D_1^kD_2)^{-1}V^{(k)}RD_1^kD_2D_2^{-1}D_1^{-k}D^kU \\ &= (D_1^kD_2)^{-1}V^{(k)}RD_1^kD_2D_2^{-1}D_1^{-k}D^kD_2\tilde{U} \\ &= (D_1^kD_2)^{-1}V^{(k)}RD_1^kD_2(|D|^{-1}D)^{-k}D^k\tilde{U} \\ &= (D_1^kD_2)^{-1}V^{(k)}RD_1^kD_2|D|^k\tilde{U}. \end{aligned}$$

Deste modo, atendendo ao Lema (6.1), podemos determinar a decomposição QR de  $T^{(k)}$ ,  $k = 0, 1, 2, \dots$

$$\begin{aligned} Q^{(k+1)} &= (P^{(k)})^{-1}P(k+1) \\ R^{(k+1)} &= U^{(k+1)}(U^{(k)})^{-1}. \end{aligned}$$

Ou seja,

$$\begin{aligned} Q^{(k+1)} &= D_2^{-1}D_1^{-k}(O^{(k)})^{-1}Q^{-1}QO^{(k+1)}D_1^{(k+1)}D_2 \\ &= D_2^{-1}D_1^{-k}(O^{(k)})^{-1}O^{(k+1)}D_1^{(k+1)}D_2 \end{aligned}$$

e

$$\begin{aligned} R^{(k+1)} &= D_2^{-1}D_1^{-k-1}V^{(k+1)}RD_1^{k+1}D_2|D|^{k+1}\tilde{U}\tilde{U}^{-1}|D|^{-k}D_2^{-1}D_1^{-k}R^{-1}(V^{(k)})^{-1}D_1^kD_2 \\ &= D_2^{-1}D_1^{-k-1}V^{(k+1)}RD_1|D|R^{-1}(V^{(k)})^{-1}D_1^kD_2. \end{aligned}$$

Como  $O^{(k)} \rightarrow I_n$  então  $(O^{(k)})^{-1}O^{(k+1)} \rightarrow I_n$  logo, sendo  $D_1^{-k}$  e  $D_1^{k+1}$  limitadas<sup>3</sup>, então

$$D_1^{-k}(O^{(k)})^{-1}(O^{(k+1)})D_1^{k+1} \rightarrow D_1$$

e, portanto,

$$Q^{(k)} \rightarrow D_2^{-1}D_1D_2 = D_1.$$

De forma semelhante, como  $V^{(k)} \rightarrow I_n$ ,

$$R^{(k+1)} \rightarrow D_2^{-1}D_1^{-k}RDR^{-1}D_1^{k-1}D_2 = \tilde{R}^{(k+1)}.$$

---

<sup>3</sup>Note-se que  $\frac{\lambda_i}{|\lambda_i|} = \pm 1, i = 1, \dots, n$ , pois  $\lambda_i \neq 0$ .

De notar que  $\tilde{R}^{(k+1)}$  é triangular superior e que  $Q^{(k+1)}$  tende para uma matriz diagonal. Assim,  $T^{(k)}$  tende para uma matriz triangular superior. Só nos resta mostrar que a diagonal de  $T^{(k)}$  tende para a diagonal de  $D$ . Para tal basta ter em conta que a diagonal de  $\tilde{R}^{(k+1)}$  é a diagonal de  $DD_1^{-1}$ , isto é, é a diagonal de  $DD^{-1}|D|$ , ou seja, de  $|D|$ .

Assim, a diagonal de  $T^{(k)} = Q^{(k+1)}\tilde{R}^{(k+1)}$  converge para a diagonal de  $D_1|D|$ , ou seja, a diagonal de  $T^{(k)}$  tende para  $D$ .  $\square$

### Observações:

1. Sob as condições do Teorema 6.4, a matriz  $T^{(0)}$  também pode ter a forma superior de Hessenberg, ou seja, o teorema também fornece condições de convergência para o Método de Hessenberg-QR.
2. De notar que, tal como refere [29, p. 519], a questão da matriz  $Y$  admitir fatorização LU pode ser ultrapassada usando uma estratégia de pivotagem introduzindo, para tal, uma matriz de permutação. Como este procedimento altera ainda significativamente a prova apresentada, tornando-a mais elaborada e extensa, optámos por não a incluir. De qualquer forma, de seguida apresentamos uma versão generalizada do Teorema 6.4 que constitui o resultado clássico mais frequente na literatura sobre convergência do Método QR.

**Teorema 6.5** *Seja  $A \in \mathbb{R}^{n \times n}$  uma matriz invertível cujos valores próprios,  $\lambda_1, \dots, \lambda_n$ , são reais e distintos em módulo. Então*

$$\lim_{k \rightarrow \infty} T^{(k)} = \begin{bmatrix} \lambda_1 & * & \cdots & \cdots & * \\ 0 & \lambda_2 & \ddots & & \vdots \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \vdots & & \ddots & \ddots & * \\ 0 & \cdots & \cdots & 0 & \lambda_n \end{bmatrix}$$

([29, pp. 521]).

3. Note-se que se  $A$  não é invertível a aplicação do Método Iterativo QR é também possível e a convergência, sob condições análogas às do Teorema 6.4, é garantida. Para tal, devemos partir da forma superior de Hessenberg da matriz e executar a iteração QR complementada com técnicas de decomposição por blocos das matrizes sucessivamente obtidas. Para mais pormenor consultar [22, p. 175].

## 6.3 O Método de Jacobi

Assim como o método QR, também o Método de Jacobi devolve uma aproximação de todo o espectro de uma dada matriz  $A$ , neste caso, real e simétrica.

Nesta secção explanaremos o Método de Jacobi, seguindo os autores [20, pp. 226–229], [22, pp. 180–184], [29, pp. 266–280] e [13, pp. 328–333]. De uma forma sucinta, o Método de Jacobi gera uma sucessão de matrizes ortogonalmente semelhantes à matriz  $A$  que converge para uma matriz diagonal, em que os elementos da diagonal são os valores próprios de  $A$ .

Consideremos  $A \in \mathbb{R}^{n \times n}$  uma matriz simétrica. O Método de Jacobi começa por tomar  $D^{(0)} = A$  e, de seguida, gera uma sucessão de matrizes

$$D^{(k)} = G_k^T D^{(k-1)} G_k,$$

onde  $G_k, k > 1$ , é uma matriz de Givens (Definição (2.18)), que irá convergir para uma matriz diagonal  $D = \text{diag}(\lambda_1, \dots, \lambda_n)$ .

Em cada iteração construímos uma matriz de Givens  $G_k \equiv G_k(p, q, \theta_k)$ , com  $p \neq q$ , de tal forma que se anulam os elementos  $d_{pq}^{(k)}$  e  $d_{qp}^{(k)}$ .

Vamos começar por verificar como construir as matrizes  $G_k$  para, de seguida, introduzirmos a forma de escolha dos índices  $p$  e  $q$ . Finalmente, verificaremos a convergência do método.

Tomemos, por exemplo, a iteração  $k$

$$D^{(k)} = G_k^T D^{(k-1)} G_k, \quad (6.31)$$

em que  $G_k \equiv G_k(p, q, \theta_k)$ , com  $p \neq q$  e  $\theta_k$  é tal forma que  $d_{pq}^{(k)} = d_{qp}^{(k)} = 0$ . A operação (6.31) apenas altera as linhas e as colunas  $p$  e  $q$  da matriz  $D^{(k-1)}$  e os elementos da matriz  $D^{(k)}$  são da seguinte forma

$$\begin{aligned} d_{ip}^{(k)} = d_{pi}^{(k)} &= \cos(\theta_k) d_{ip}^{(k-1)} + \sin(\theta_k) d_{iq}^{(k-1)}, \quad i \neq p, q \\ d_{iq}^{(k)} = d_{qi}^{(k)} &= \cos(\theta_k) d_{iq}^{(k-1)} - \sin(\theta_k) d_{ip}^{(k-1)}, \quad i \neq p, q \\ d_{pp}^{(k)} &= \cos(\theta_k)^2 d_{pp}^{(k-1)} + 2 \cos(\theta_k) \sin(\theta_k) d_{pq}^{(k-1)} + \sin(\theta_k)^2 d_{qq}^{(k-1)} \\ d_{qq}^{(k)} &= \sin(\theta_k)^2 d_{pp}^{(k-1)} - 2 \cos(\theta_k) \sin(\theta_k) d_{pq}^{(k-1)} + \cos(\theta_k)^2 d_{qq}^{(k-1)} \\ d_{pq}^{(k)} &= (\cos(\theta_k)^2 - \sin(\theta_k)^2) d_{pq}^{(k-1)} + \cos(\theta_k) \sin(\theta_k) (d_{qq}^{(k-1)} - d_{pp}^{(k-1)}). \end{aligned} \quad (6.32)$$

Vamos fazer  $c := \cos(\theta_k)$  e  $s := \sin(\theta_k)$ . Lembramos que

$$\cos(2\theta_k) = \cos(\theta_k)^2 - \sin(\theta_k)^2 = c^2 - s^2$$

e

$$\sin(2\theta_k) = 2 \cos(\theta_k) \sin(\theta_k) = 2cs$$

donde

$$\cot(2\theta_k) = \frac{c^2 - s^2}{2cs}. \quad (6.33)$$

Suponhamos  $d_{pq}^{(k-1)} \neq 0$ , caso contrário basta tomar  $\cos(\theta_k) = 1$  e  $\sin(\theta_k) = 0$ . Pretendemos que o elemento  $d_{pq}^{(k)}$  seja nulo, ou seja,

$$\begin{aligned} (c^2 - s^2) d_{pq}^{(k-1)} + cs(d_{qq}^{(k-1)} - d_{pp}^{(k-1)}) &= 0 \\ \frac{c^2 - s^2}{cs} &= \frac{d_{pp}^{(k-1)} - d_{qq}^{(k-1)}}{d_{pq}^{(k-1)}}. \end{aligned}$$

Tome-se

$$a := \frac{d_{pp}^{(k-1)} - d_{qq}^{(k-1)}}{2d_{pq}^{(k-1)}}. \quad (6.34)$$

Então a equação (6.33) fica

$$\cot(2\theta_k) = a.$$

Daqui podemos extrair o valor de  $\theta_k$ ,

$$\theta_k = \frac{1}{2} \operatorname{arccot}(a).$$

No entanto, e de forma a reduzir os erros de arredondamento, podemos resolver a situação de um outro modo. Seja

$$t := \tan(\theta_k) = \frac{c}{s}.$$

Retomemos a equação (6.33),

$$\begin{aligned} a &= \frac{c^2 - s^2}{2cs} \\ a &= \frac{\frac{c^2 - s^2}{c^2}}{\frac{2cs}{c^2}} \\ a &= \frac{1 - t^2}{2t} \\ 2at &= 1 - t^2 \\ t^2 + 2at - 1 &= 0 \\ t &= \frac{-2a \pm \sqrt{4a^2 + 4}}{2} \\ t &= -a \pm \sqrt{a^2 + 1}. \end{aligned} \quad (6.35)$$

Uma das escolhas para a raiz da equação anterior, que garante a convergência do método, é tomar  $|t| \leq 1$ , o que implica  $|\theta_k| \leq \frac{\pi}{4}$ . Em (6.35), se  $a \geq 0$  escolhemos o sinal positivo, caso contrário escolhemos o sinal negativo, ou seja,

$$\begin{aligned} t &= \operatorname{sinal}(a) \left( -|a| + \sqrt{a^2 + 1} \right) \\ \left( |a| + \sqrt{a^2 + 1} \right) t &= \operatorname{sinal}(a) \left( -|a| + \sqrt{a^2 + 1} \right) \left( |a| + \sqrt{a^2 + 1} \right) \\ \left( |a| + \sqrt{a^2 + 1} \right) t &= \operatorname{sinal}(a) \\ t &= \frac{\operatorname{sinal}(a)}{|a| + \sqrt{a^2 + 1}}. \end{aligned} \quad (6.36)$$

Finalmente, e recordando que  $t = \frac{c}{s} = \frac{\sqrt{1-c^2}}{c}$ , obtemos

$$c = \frac{1}{\sqrt{t^2 + 1}}$$

e

$$s = tc,$$

o que permite determinar, em cada iteração, a matriz de Givens correspondente, tomando  $a$  tal como em (6.34) e determinando  $t$  tal como em (6.36).

Por um processo análogo a (6.23), facilmente constatamos que o Método de Jacobi devolve uma sucessão de matrizes  $D^{(k)}$  ortogonalmente semelhante à matriz inicial,  $A$ .

Salientamos que as sucessivas transformações não preservam as operações anteriores, ou seja, cada iteração anula um elemento fora da diagonal mas não preserva o elemento anteriormente anulado. No entanto, ao longo do processo os elementos fora da diagonal tendem a ficar muito próximos de zero.

Sendo o objetivo do processo o anulamento dos elementos fora da diagonal da matriz inicial há que decidir por onde começar, ou seja, qual o primeiro elemento de  $D^{(0)} = A$  a anular. Assim, o Método Clássico de Jacobi inicia cada iteração procurando o elemento, fora da diagonal principal, com maior valor absoluto. No entanto, este processo pode tornar-se moroso.

Uma outra forma de escolher os índices é optar, sucessivamente, pelos elementos acima da diagonal principal, ou seja, fazer

$$(p, q) = (1, 2), (1, 3), \dots, (1, n); (2, 3), (2, 4), \dots, (2, n); \dots; (n-1, n)$$

e retomar o processo [29, p. 269].

**Exemplo 6.12** Voltamos à matriz (4.11). Na aplicação do método, a este exemplo, seguimos os seguintes passos:

- escolhemos como pivot o elemento fora da diagonal com maior valor absoluto;
- calculámos  $a = \frac{d_{pp}^{(k-1)} - d_{qq}^{(k-1)}}{2d_{pq}^{(k-1)}}$ ;
- calculámos  $t = \frac{\sin(a)}{|a| + \sqrt{a^2 + 1}}$ ;
- fizemos  $c = \frac{1}{\sqrt{t^2 + 1}}$  e  $s = tc$ ;
- construímos a matriz de rotação  $G_k$  fazendo  $\cos(\theta_k) = c$  e  $\sin(\theta_k) = s$
- fizemos  $D^{(k)} = G_k^T D^{(k-1)} G_k, k \geq 1$ , com  $D^{(0)} = A$ .

e após 20 iterações obtivemos

$$D^{(20)} \cong \begin{bmatrix} -2.563382668195001 & 0.000000000000000 & 0 & -0.000000000000000 \\ 0.000000000000000 & -0.295188571810782 & -0.000000000000000 & 0.000000000000000 \\ 0.000000000000000 & 0.000000000000000 & 11.840474193588960 & -0.000000000000000 \\ 0.000000000000000 & 0.000000000000000 & 0.000000000000000 & 4.018097046416819 \end{bmatrix}$$



donde podemos extrair aproximações para os valores próprios da matriz  $A$ ,

$$\begin{aligned}\lambda_1 &\cong -2.563382668195001 \\ \lambda_2 &\cong -0.295188571810782 \\ \lambda_3 &\cong 11.840474193588960 \\ \lambda_4 &\cong 4.018097046416819.\end{aligned}$$

### 6.3.1 Convergência do Método de Jacobi

Resta-nos verificar que, qualquer que seja a nossa forma de escolher o elemento  $d_{pq}^{(k)}$  a anular, o método de Jacobi converge. Apresentamos a prova da convergência do método escolhendo o método clássico, ou seja, quando escolhemos o elemento fora da diagonal com maior valor absoluto, seguindo [22, pp. 182–183] e [19, pp. 409–410].

**Teorema 6.6** *O Método Clássico de Jacobi , com  $|\theta_k| \leq \frac{\pi}{4}$ , converge.*

**Prova:** Suponhamos que optámos pelo Método Clássico de Jacobi, ou seja, escolhemos o elemento fora da diagonal com maior valor absoluto. Vamos ter em conta que

$$|\theta_k| \leq \frac{\pi}{4}$$

e vamos decompor a matriz  $D^{(k)}$ , com  $k \geq 1$ , na sua parte diagonal,  $\tilde{D}^{(k)}$ , e na sua parte não diagonal,  $E^{(k)}$ . Então temos

$$D^{(k)} = \tilde{D}^{(k)} + E^{(k)}.$$

Para cada rotação do Método de Jacobi,  $G_k$ , e usando a norma de Frobenius, (ver página 13), temos

$$\begin{aligned}\|E^{(k)}\|_F^2 &= \sum_{\substack{i,j=1 \\ i \neq j}}^n (d_{ij}^{(k)})^2 \\ &= \sum_{\substack{i,j=1 \\ i \neq j \neq p,q}}^n \left[ (d_{ij}^{(k)})^2 + 2(d_{ip}^{(k)})^2 + 2(d_{iq}^{(k)})^2 \right] + 2(d_{pq}^{(k)})^2.\end{aligned}$$

Relembramos que as equações (6.32) mostram como podemos escrever os elementos de  $D^{(k)}$  em função dos elementos de  $D^{(k-1)}$ , que os elementos  $d_{ij}^{(k)} = d_{ij}^{(k-1)}$ , com  $i \neq j \neq p, q$ ,

e que o elemento  $d_{pq}^{(k)} = 0$ . Assim,

$$\begin{aligned}
 \|E^{(k)}\|_F^2 &= \sum_{\substack{i,j=1 \\ i \neq j \neq p,q}}^n (d_{ij}^{(k-1)})^2 + 2 \sum_{\substack{i=1 \\ i \neq p,q}}^n [(cd_{ip}^{k-1} + sd_{iq}^{k-1})^2 + (cd_{iq}^{k-1} + sd_{ip}^{k-1})^2] \\
 &= \sum_{\substack{i,j=1 \\ i \neq j \neq p,q}}^n (d_{ij}^{(k-1)})^2 + 2 \sum_{\substack{i=1 \\ i \neq p,q}}^n [c^2(d_{ip}^{k-1})^2 + 2csd_{ip}^{k-1}d_{iq}^{k-1} + s^2(d_{iq}^{k-1})^2 + c^2(d_{iq}^{k-1})^2 \\
 &\quad + 2csd_{ip}^{k-1}d_{iq}^{k-1} + s^2(d_{ip}^{k-1})^2] \\
 &= \sum_{\substack{i,j=1 \\ i \neq j \neq p,q}}^n (d_{ij}^{(k-1)})^2 + 2 \sum_{\substack{i=1 \\ i \neq p,q}}^n [(c^2 + s^2)(d_{ip}^{k-1})^2 + (c^2 + s^2)(d_{iq}^{k-1})^2].
 \end{aligned}$$

Como  $c^2 + s^2 = 1$ ,

$$\begin{aligned}
 \|E^{(k)}\|_F^2 &= \sum_{\substack{i,j=1 \\ i \neq j \neq p,q}}^n (d_{ij}^{(k-1)})^2 + 2 \sum_{\substack{i=1 \\ i \neq p,q}}^n [(d_{ip}^{k-1})^2 + (d_{iq}^{k-1})^2] \\
 &= \sum_{\substack{i,j=1 \\ i \neq j \neq p,q}}^n (d_{ij}^{(k-1)})^2 + 2 \sum_{\substack{i=1 \\ i \neq p,q}}^n [(d_{ip}^{k-1})^2 + (d_{iq}^{k-1})^2] + 2(d_{pq}^{(k-1)})^2 - 2(d_{pq}^{(k-1)})^2 \\
 &= \sum_{\substack{i,j=1 \\ i \neq j}}^n (d_{ij}^{(k-1)})^2 - 2(d_{pq}^{(k-1)})^2 \\
 &= \|E^{(k-1)}\|_F^2 - 2(d_{pq}^{(k-1)})^2,
 \end{aligned} \tag{6.37}$$

e  $d_{pq}^{(k-1)} \neq 0$ , a cada iteração.

Como já referimos, ao longo das iterações, os elementos fora da diagonal vão decrescendo, ou seja, vão ficando cada vez mais próximos de zero, em valor absoluto, logo  $\|E^{(k)}\|_F^2$  também decresce com  $k$  ao longo da aplicação do método. Seja  $N = n(n-1)$  o número de elementos fora da diagonal principal da matriz  $A$ . Como escolhemos para pivot o elemento

$$|d_{pq}^{(k)}| \geq |d_{ij}^{(k)}|, \forall i, j = 1, 2, \dots, n,$$

então

$$\|E^{(k)}\|_F^2 = \sum_{\substack{i,j=1 \\ i \neq j}}^n (d_{ij}^{(k)})^2 \leq N|d_{pq}^{(k)}|^2.$$

Podemos ver que

$$\begin{aligned}
 \|E^{(k+1)}\|_F^2 &= \|E^{(k)}\|_F^2 - 2(d_{pq}^{(k)})^2, \text{ por (6.37)} \\
 &\leq \|E^{(k)}\|_F^2 - 2 \frac{\|E^{(k)}\|_F^2}{N} \\
 &= \left(1 - \frac{2}{N}\right) \|E^{(k)}\|_F^2, \forall k
 \end{aligned}$$

então, fazendo  $\rho := 1 - \frac{2}{N}$ ,

$$\begin{aligned}
 \|E^{(k)}\|_F^2 &\leq \rho \|E^{(k-1)}\|_F^2 \\
 &\leq \rho (\rho \|E^{(k-2)}\|_F^2) \\
 &\leq \rho^2 (\rho \|E^{(k-3)}\|_F^2) \\
 &\vdots \\
 &\leq \rho^k \|E^{(0)}\|_F^2
 \end{aligned} \tag{6.38}$$

donde  $\|E^{(k)}\|_F^2$  tende para zero, ou seja, as matrizes  $D^{(k)}$ , geradas pelo método clássico de Jacobi, tendem a ter elementos, fora da diagonal principal, nulos.

Vamos agora verificar o que é que acontece aos elementos da diagonal principal das matrizes  $D^{(k)}, \forall k$ .

Ao longo das iterações fazemos

$$(c^2 - s^2)d_{pq}^{(k-1)} + cs(d_{qq}^{(k-1)} - d_{pp}^{(k-1)}) = 0$$

donde

$$d_{qq}^{(k-1)} = d_{pp}^{(k-1)} - \frac{c^2 - s^2}{cs} d_{pq}^{(k-1)}.$$

Podemos reescrever os elementos  $d_{pp}^{(k)}$  da seguinte forma

$$\begin{aligned}
 d_{pp}^{(k)} &= c^2 d_{pp}^{(k-1)} + 2cs d_{pq}^{(k-1)} + s^2 d_{qq}^{(k-1)} \\
 &= c^2 d_{pp}^{(k-1)} + 2cs d_{pq}^{(k-1)} + s^2 \left( d_{pp}^{(k-1)} - \frac{c^2 - s^2}{cs} d_{pq}^{(k-1)} \right) \\
 &= (c^2 + s^2) d_{pp}^{(k-1)} + \left( 2cs - \frac{c^2 - s^2}{cs} \right) d_{pq}^{(k-1)} \\
 &= d_{pp}^{(k-1)} + t_k d_{pq}^{(k-1)}, \forall k,
 \end{aligned}$$

com  $t_k = \tan(\theta_k)$ . Então

$$d_{pp}^{(k)} - d_{pp}^{(k-1)} = t_k d_{pq}^{(k-1)}.$$

Como, por hipótese,  $|\theta_k| \leq \frac{\pi}{4}$ , temos que  $|t_k| \leq 1$ , donde

$$|d_{pp}^{(k)} - d_{pp}^{(k-1)}| \leq |d_{pq}^{(k-1)}|, \forall k.$$

Do mesmo modo,

$$|d_{qq}^{(k)} - d_{qq}^{(k-1)}| \leq |d_{pq}^{(k-1)}|, \forall k.$$

As outras entradas, na iteração  $k$ , não são alteradas, logo

$$\|\tilde{D}^{(k+1)} - \tilde{D}^{(k)}\|_F \leq \|E^{(k)}\|_F.$$

Então

$$\begin{aligned}
 \left\| \tilde{D}^{(\ell)} - \tilde{D}^{(k)} \right\|_F &\leq \left\| \tilde{D}^{(\ell)} - \tilde{D}^{(\ell-1)} \right\|_F + \left\| \tilde{D}^{(\ell-1)} - \tilde{D}^{(\ell-2)} \right\|_F + \dots + \left\| \tilde{D}^{(k+1)} - \tilde{D}^{(k)} \right\|_F \\
 &\leq \|E^{(\ell-1)}\|_F + \|E^{(\ell-2)}\|_F + \dots + \|E^{(k)}\|_F \\
 &\leq \rho^{\ell-1} \|E^{(0)}\|_F + \dots + \rho^k \|E^{(0)}\|_F, \text{ por (6.38)} \\
 &= (\rho^{\ell-1} + \dots + \rho^k) \|E^{(0)}\|_F \\
 &= \frac{\rho^k}{1 - \rho} \|E^{(0)}\|_F, \ell > k.
 \end{aligned}$$

Assim, como  $\rho^k \rightarrow 0$ , pois  $|\rho| < 1$ , a sucessão de matrizes  $\tilde{D}^{(k)}$ , com  $k \in \mathbb{N}$ , é uma sucessão de Cauchy<sup>4</sup>, logo convergente. Seja  $D$  tal que  $\tilde{D}^{(k)} \rightarrow D$ . Como  $E^{(k)} \rightarrow 0$  então  $D^{(k)} \rightarrow D$ . A matriz  $D$  é diagonal e tem o mesmo espectro que  $A$  pois, em cada iteração, as matrizes  $D^{(k)}$  são-lhe ortogonalmente semelhantes.  $\square$

---

<sup>4</sup>Uma sucessão  $x_{n,n \in \mathbb{N}}$  diz-se de Cauchy se para todo o  $\varepsilon > 0$  existe um número inteiro  $N$  tal que  $|x_m - x_n| \leq \varepsilon$ , com  $m, n \geq N$ .

# Capítulo 7

## Conclusões

### 7.1 Síntese do Trabalho Apresentado

A principal conclusão desta dissertação terá de ser o facto de nos termos apercebido de uma vasta área de estudo, muito antiga mas em constante desenvolvimento e que tem importantíssimas aplicações em variadas áreas das ciências e da engenharia. Por razões históricas, começámos por formular o problema como o cálculo de raízes de polinómios. Esse problema tem uma história milenar e acompanhou as várias civilizações do Homem. A Álgebra nasceu para tratar este problema. Tratámos, por isso, métodos muito antigos de localização de zeros de polinómios (ver Apêndice A). Também por isso, tratámos métodos que permitem construir o polinómio caraterístico a partir da matriz dada: o Método de Danilevsky, o Método de Krylov, o Método de Leverrier e o Método dos coeficientes indeterminados.

Estes métodos são interessantes porque tentam encontrar o processo mais eficiente para construir o polinómio caraterístico. Nesse contexto, devemos referir que o Método de Danilevsky é o preferível, se a ordem da matriz for superior a cinco. O desafio de encontrar métodos mais eficientes subsiste. Contudo, este assunto é só um subproblema do problema geral de calcular os valores próprios de uma matriz. Formado o polinómio, há que calcular as suas raízes. Para isso apresentámos alguns métodos antigos, conjuntamente com o método de Bairstow: o Método de Newton-Horner, que é muito adaptado ao cálculo manual; o Método de Müller, que é uma generalização dos antiquíssimos métodos da corda falsa (*“regula falsi”*) e método da secante; e o Método de Bairstow, que calcula fatores quadráticos do polinómio usando só aritmética de reais e exhibe convergência quadrática (sob certas condições).

Uma das grandes desvantagens destes métodos é que são usados em conjunto com o método da deflação. Ora, se o polinómio é de elevada ordem, o acumular de erros de arredondamento nas sucessivas deflações compromete a precisão do cálculo das raízes. Outra desvantagem surge na seleção de aproximações iniciais. Todos aqueles métodos convergem se a aproximação inicial for suficientemente boa. Muitas vezes não conseguimos boas aproximações para todas as raízes. Os dois primeiros métodos podem ser adaptados

ao cálculo de raízes complexas mas exigem que os cálculos sejam feitos com aritmética de complexos. Por todas estas razões, os melhores métodos são os métodos iterativos matriciais. Contudo, é sabido que estes métodos só tiveram ampla aplicação com o advento dos computadores electrónicos digitais após 1950.

Os métodos matriciais são especialmente poderosos porque é possível vetorizar o código dos programas de computador de forma muito eficiente. Além disso, a teoria de matrizes está muito desenvolvida e tem ferramentas eficazes para analisar os métodos. Adicionalmente, podemos referir que estes métodos são muito fáceis de perceber, na sua formulação base, e também muito fáceis de programar. Dedicámos, assim, o devido espaço aos seguintes métodos iterativos matriciais: Método da Potência, Método QR e Método de Jacobi.

Os métodos iterativos matriciais podem calcular um só valor próprio, como é o caso do Método da Potência na sua formulação base. Podem também aplicar-se a classes especiais de matrizes, de que é exemplo o Método de Jacobi que se aplica a matrizes simétricas. Contudo o Método QR calcula todo o espectro de uma matriz arbitrária de elementos reais e pode ser adaptado para matrizes de entradas complexas. Por essas razões, alguns autores classificam o Método QR como um *método geral*. Por oposição, os outros métodos são designados por *métodos parciais*.

Além de se descrever os métodos de forma consistente, procedemos à análise da sua convergência. As análises de convergência dos Métodos das Potências e do Método QR, que incluímos, foram feitas assumindo que estas matrizes eram diagonalizáveis ou invertíveis, respetivamente. Esta suposição atravessa quase toda a literatura que encontrámos. Na verdade, a sua supressão torna essa análise mais difícil. Apresentámos exemplos numéricos que atravessam todo o texto. Esses exemplos foram obtidos após simulações de programas MATLAB®, especialmente programados por nós, e que se encontram em apêndice. As vantagens e desvantagens dos vários métodos foram brevemente discutida. Terminamos referindo que o nosso estudo nos deu uma perspectiva não só histórica do problema em causa como nos levou ao estudo dos métodos mais poderosos que se usam na atualidade.

## 7.2 Pistas para Trabalho Adicional

Como já foi dito, dada a vastidão do tema abordado houve que fazer opções e escolhas para limitar a extensão desta dissertação. Além disso, optámos por uma abordagem inclinada para os aspetos de consistência e convergência dos métodos e os exemplos estudados foram sempre de matrizes de pequena dimensão. Julgamos não poder terminar esta dissertação sem antes apontar algumas pistas para trabalho adicional que foram vislumbradas na execução do mesmo que conduziu a esta dissertação.

Este mesmo tema de trabalho (*“The Matrix Eigenvalue Problem”*), poderia ter sido abordado numa vertente mais computacional (e ligado às aplicações), se tivéssemos dedicado o devido espaço aos seguintes tópicos:

- Problemas de matrizes simétricas esparsas de grandes dimensão. Teriam de ser tratadas as importantes preocupações inerentes de eficiência computacional e poupança

de espaço de memória. Neste contexto, teria de se estudar o método de Lanczos, por ser muito popular nestes casos.

- Problemas com matrizes “zero e um”, que são matrizes cujas entradas só podem ser zero ou um. Essas matrizes têm aplicações específicas. Com efeito, a matriz de adjacência de um grafo é desse tipo e a teoria espectral de grafos necessita do cálculo eficiente e preciso dos seus valores próprios.
- Estudo das especificidades ligadas às aplicações (ao algoritmo “*PageRank*”, por exemplo), que pode estar relacionado, por exemplo, com os pontos anteriores;
- Técnicas de “shifting” ou outras, que permitam lidar com matrizes de espectro mal separado, que provocam, em geral, problemas de convergência. Esses problemas podem ser “apenas” de lentidão de convergência ou falha completa desta;
- Análise do condicionamento do problema de calcular os valores próprios;
- Análises de erro de truncatura e estudo de condições de paragem dos algoritmos;
- Análise da propagação dos erros de arredondamento, ou seja, da estabilidade numérica dos métodos. Neste ponto é essencial tratar as técnicas de pivotagem para evitar as divisões por “*pivots*” de baixo valor absoluto.
- Análise da ordem de convergência dos algoritmos;
- Análise comparativa de simulações computacionais. Neste tópico, podiam-se comparar os vários métodos em termos de velocidade de processamento *versus* precisão dos resultados obtidos. Na internet encontram-se matrizes teste de grande dimensão que poderiam ser usadas nos ensaios computacionais.

Outra pista incontornável está ligada ao tratamento necessariamente incompleto que fizemos do Método QR. Não se trataram todas as suas variantes nem se provou a sua convergência com a máxima generalidade. As questões relativas à ordem (e velocidade) de convergência também não puderam ter o desenvolvimento devido, nesta dissertação, por limitações de espaço. Estamos convencidos que o estudo do Método QR constituiria por si só trabalho suficiente para outra monografia separada, incluindo trabalho de programação e de ensaios computacionais. O método é atual, muito utilizado, com aspetos teóricos interessantes e de fácil compreensão. Está também em constante desenvolvimento, sendo ainda hoje matéria de investigação, ([26]).

Temos também de referir que a ênfase da presente dissertação foi no cálculo dos valores próprios. Muitas vezes não se referiram as possibilidades de calcular os correspondentes vetores próprios. Ora, nas aplicações poderemos estar interessados no cálculo dos pares próprios. Nesses casos, convém não separar os dois problemas tentando métodos que permitam obter pares próprios. Em alternativa, poder-se-ia estudar métodos de cálculo de vetores próprios *a posteriori*.

Outra pista para trabalho adicional é a generalização natural dos métodos para matrizes de entradas complexas. Não só o Método QR mas outros métodos poderiam ser estudados, havendo especificidades a tratar e adaptações a ter em conta.

Por último, teremos de referir que o problema do cálculo do espectro de um operador pode ser generalizado para espaços de dimensão infinita. A primeira abordagem deve-se a Hilbert que considerou operadores integrais relacionando-os com matrizes infinitas. Esta última pista levaria o nosso trabalho da área da Álgebra Linear Numérica para o campo mais geral dos espaços de funções, teoria dos operadores e Análise Funcional. Ou seja, o “*The Matrix Eigenvalue Problem*” poderia ser generalizado para o “*The Operator Eigenvalue Problem*.”

---



# Apêndice A

## Localização de Zeros de Polinômios

No Capítulo 3 vimos alguns processos de localização de valores próprios. Uma outra forma de localizarmos os valores próprios de uma dada matriz é localizando as raízes do seu polinômio característico.

Uma primeira forma de resolver o problema da localização destas raízes pode passar pela observação do gráfico do polinômio em estudo. No entanto, só teremos acesso ao número de raízes reais e num determinado domínio. Detetado um intervalo com uma raiz, poderemos aplicar o Teorema de Bolzano para mostrar a existência desse zero. Este teorema pode ser aplicado, visto que qualquer polinômio é uma função contínua.

Neste apêndice, veremos alguns métodos de determinação do número de raízes reais de um polinômio e depois abordaremos alguns métodos que nos permitem localizá-las.

### A.1 Número de raízes reais de um polinômio

Apresentaremos, de seguida, dois métodos de determinação do número de zeros reais de um polinômio. Estas regras permitem provar a existência de um certo número de raízes em  $\mathbb{R}$  ou em certos subconjuntos de  $\mathbb{R}$ . A sua aplicação permite começar a reunir informação acerca da localização das raízes de um polinômio.

Suponhamos um polinômio com coeficientes reais. Então se este possuir raízes complexas estas só podem surgir aos pares conjugados, ou seja, as raízes complexas do polinômio  $p(x)$  serão em número par. Daqui podemos afirmar que se um polinômio  $p(x)$  for de grau  $n$ , ímpar, então terá um número ímpar de raízes reais. Por outro lado, se um polinômio  $p(x)$  for de grau  $n$ , par, o número de raízes reais será par.

Vamos considerar um polinômio de grau  $n$  escrito da seguinte forma:

$$p(x) = a_n x^n + a_{n-1} x^{n-1} + \cdots + a_1 x + a_0. \quad (\text{A.1})$$

onde  $a_i \in \mathbb{R}, i = 0, \dots, n$ , e  $a_n \neq 0$ .

**Definição A.1** *Tendo em conta o polinômio (A.1) vamos designar por  $V$  o número de*

variações de sinal dos coeficientes, não nulos, do polinómio, por  $N_+$  o número de zeros reais positivos e por  $N_-$  o número de zeros reais negativos.

### A.1.1 Regra de Budan-Fourier

Esta regra aplica-se a um intervalo aberto e limitado, podendo ficar a saber-se o número de zeros reais nesse mesmo intervalo.

Sejam  $a, b \in \mathbb{R}$  e tomemos em consideração as seguintes sucessões:

$$p(a), p'(a), p''(a), \dots, p^{(n)}(a) \quad , \text{ com } p(a) \neq 0$$

$$p(b), p'(b), p''(b), \dots, p^{(n)}(b) \quad , \text{ com } p(b) \neq 0.$$

Seja  $V_a$  o número de variações de sinal da primeira sucessão e  $V_b$  o número de variações de sinal da segunda.

**Teorema A.1** *O número  $N$  de zeros reais do polinómio  $p$  no intervalo  $]a, b[$  não excede o valor  $V_a - V_b$  e  $(V_a - V_b) - N$  é par.*

A prova deste teorema pode ser consultada em [14, pp. 244–246].

### A.1.2 Regra dos Sinais de Descartes

A Regra dos Sinais de Descartes relaciona o número de raízes reais com o número de variações de sinal dos coeficientes do polinómio em estudo. De seguida, enunciamos o teorema correspondente e a sua prova pode ser consultada em [14, pp. 247–249] ou [25].

**Teorema A.2 (Regra de Sinais de Descartes)** *Seja  $p$  um polinómio de coeficientes reais. Então:*

- a) *o número  $N_+$  de  $p$  não excede o número  $V$ , de  $p$ , e o valor  $V - N_+$  é par;*
- b) *o número  $N_-$  de  $p$  não excede o número  $V$ , de  $q(x) = p(-x)$ , e o valor  $V - N_-$  é par.*

Esta regra, que é uma regra generalizada da Regra de Boudan-Fourier, permite contar os zeros reais em intervalos da forma  $I_1 = ]-\infty, a[$  e  $I_2 = ]a, +\infty[$ .

### A.1.3 Exemplo

**Exemplo A.1** *Vamos voltar ao Exemplo 4.1 em que se chegou ao polinómio caraterístico*

$$p(x) = x^4 - 13x^3 + 3x^2 + 124x + 36.$$

*Obtivemos um polinómio de grau 4 que, pelo Teorema Fundamental da Álgebra, possui 4 raízes, reais ou complexas, contando com as suas multiplicidades. Como  $p(x)$  é de grau 4, ou seja par, então terá um número par de raízes reais, neste caso, zero, duas ou quatro.*

*Vamos aplicar as diferentes regras enunciadas anteriormente:*

	Sinal em $a = -4$	Sinal em $b = 5$
$p(x) = x^4 - 13x^3 + 3x^2 + 124x + 36$	+	−
$p'(x) = 4x^3 - 39x^2 + 6x + 124$	−	−
$p''(x) = 12x^2 - 78x + 6$	+	−
$p'''(x) = 24x - 78$	−	+
$p^{(4)}(x) = 24$	+	+
	$V_{-4} = 4$	$V_5 = 1$

Tabela A.1: Aplicação da regra de Budan-Fourier

### Regra de Budan-Fourier

Vamos aplicar esta regra ao intervalo  $] -4, 5[$ , pois  $p(4) \neq 0$  e  $p(5) \neq 0$ .

Pelos valores encontrados na Tabela A.1, temos que  $V_{-4} - V_5 = 4 - 1 = 3$ . Então o número,  $N$ , de raízes reais do polinómio no intervalo  $] -4, 5[$ , ou é igual a 1 ou igual a 3, porque  $V_{-4} - V_5 - 1 = 2$  é par e  $V_{-4} - V_5 - 3 = 0$  é par.

### Regra dos Sinais de Descartes

Temos  $V = 2$  donde, pela regra de sinais de Descartes,  $N_+ \leq 2$  e  $V - N_+$  é par. Então  $N_+ = 2$  ou  $N_+ = 0$ .

Faça-se  $q(x) = p(-x) = x^4 + 13x^3 + 3x^2 - 124x + 36$ . Temos  $V = 2$  donde  $N_- \leq 2$  e  $V - N_-$  é par. Então  $N_- = 2$  ou  $N_- = 0$ .

Podemos então concluir que o polinómio verifica um dos casos seguintes:

- 2 raízes reais positivas e 2 raízes reais negativas;
- 2 raízes reais positivas e 2 raízes complexas conjugadas;
- 2 raízes reais negativas e 2 raízes complexas conjugadas;
- dois pares de raízes complexas conjugadas.

## A.2 Localização de raízes de um polinómio

Iremos, agora, abordar a localização de valores próprios utilizando, para tal, o seu polinómio característico. Assim, apresentamos alguns teoremas cujas provas poderão ser vistas, respetivamente, em [19, pp. 269–270] e em [4, pp. 168–169].

**Teorema A.3 (Corolário da Regra de Cauchy)** *Vamos considerar o polinómio  $p$  com  $a_0 \neq 0$ . Então todos os zeros  $x_k$ ,  $k = 1, \dots, n$ , de  $p$  verificam:*

$$\left(1 + \max_{1 \leq k \leq n} \left| \frac{a_k}{a_0} \right| \right)^{-1} \leq |x_k| \leq 1 + \max_{0 \leq k \leq n-1} \left| \frac{a_k}{a_n} \right|.$$

**Teorema A.4 (Regra de Lagrange)** *Seja  $p$  um polinómio real cujos coeficientes satisfazem as condições*

$$a_n > 0, a_{n-1} \geq 0, \dots, a_{m+1} \geq 0, a_m < 0$$

*ou seja,  $a_m$  é o primeiro coeficiente negativo na sequência  $a_n, a_{n-1}, \dots, a_0$ . Então os zeros reais de  $p$  são majorados por:*

$$z \leq 1 + \max_{a_k < 0} \left| \frac{a_k}{a_n} \right|^{\frac{1}{n-m}}$$

Da mesma forma que usamos estas regras para encontrar limites superiores para raízes positivas, podemos fazê-lo também para limitar raízes negativas. Para isso, basta aplicá-las a  $p(-x)$ . Para limitar inferiormente, aplicamos as regras a  $p\left(\frac{1}{x}\right)$ , para as raízes positivas, e a  $p\left(-\frac{1}{x}\right)$ , para as raízes negativas.

**Exemplo A.2** *Vamos voltar ao Exemplo 4.1 em que se chegou ao polinómio caraterístico*

$$p(x) = x^4 - 13x^3 + 3x^2 + 124x + 36.$$

*Vamos aplicar as regras de Cauchy e de Lagrange e vamos ter em conta que:*

$$p_1(-x) = x^4 + 13x^3 + 3x^2 - 124x + 36 \quad (\text{A.2})$$

$$p_2\left(\frac{1}{x}\right) = 36x^4 + 124x^3 + 3x^2 - 13x + 1 \quad (\text{A.3})$$

$$p_3\left(-\frac{1}{x}\right) = 36x^4 - 124x^3 + 3x^2 + 13x + 1. \quad (\text{A.4})$$

### Corolário da Regra de Cauchy

$$a_0 = 36$$

$$a_4 = 1$$

$$\max_{1 \leq k \leq 4} \left| \frac{a_k}{a_0} \right| = \max \left\{ \frac{124}{36}, \frac{3}{36}, \frac{13}{36}, \frac{1}{36} \right\} = \frac{124}{36}$$

$$\max_{0 \leq k \leq 3} \left| \frac{a_k}{a_4} \right| = \max \left\{ \frac{36}{1}, \frac{124}{1}, \frac{3}{1}, \frac{36}{1} \right\} = 124$$

*Aplicando o Corolário da regra de Cauchy, para todos os zeros  $z$  do polinómio  $p$ , vem que:*

$$\left(1 + \frac{124}{36}\right)^{-1} \leq |z| \leq 1 + 124$$

$$\frac{9}{40} \leq |z| \leq 125$$

*ou seja, todos os zeros,  $z$ , do polinómio  $p$  estão situados na coroa circular do plano complexo definida pelas circunferências de raios interior  $\frac{9}{40}$  e exterior 125, centradas na origem.*

### Regra de Lagrange

Vamos começar por aplicar a regra para localizar os zeros positivos,  $z^+$ , do polinómio  $p$ . Começamos por determinar o limite superior.

$$a_4 = 1$$

$$a_3 = -13, \text{ primeiro coeficiente negativo de } p$$

$$\max_{a_k < 0} \left| \frac{a_k}{a_4} \right| = \max \left\{ \frac{13}{1} \right\} = \frac{13}{1} = 13 \quad (\text{A.5})$$

Para determinar o limite inferior, o procedimento é o mesmo, aplicado a  $p_2$ .

$$a_4 = 36$$

$$a_1 = -13, \text{ primeiro coeficiente negativo de } p_2$$

$$\max_{a_k < 0} \left| \frac{a_k}{a_4} \right| = \max \left\{ \frac{13}{36} \right\} = \frac{13}{36} \quad (\text{A.6})$$

De (A.5) e (A.6) temos que:

$$\begin{aligned} \left(1 + \frac{13}{36}^{\frac{1}{4-1}}\right)^{-1} &\leq z^+ \leq 1 + 13 \\ 0.584 &\leq z^+ \leq 14. \end{aligned}$$

De uma forma semelhante, aplicando a regra a  $p_1$  e a  $p_3$ , temos os limites para os zeros negativos,  $z^-$ , do polinómio  $p$ :

$$0.225 \leq z^- \leq 6.$$

Como referimos, neste apêndice, uma outra forma de localizar e/ou separar raízes será a partir da aplicação do Teorema de Bolzano. Vamos elaborar uma tabela com alguns valores do polinómio. Como qualquer polinómio é uma função contínua, pelo Teorema de

$x$	$p(x)$	Variações de sinal
-3	123	
-2	-80	1
-1	-71	
0	36	2
1	151	
3	165	
5	-269	3
12	228	4

Tabela A.2: Tabela com alguns valores de  $p(x)$

*Bolzano, podemos afirmar que em cada um dos intervalos  $] -3, -2[$ ,  $] -1, 0[$ ,  $] 3, 5[$  e  $] 5, 12[$ , existe uma única raiz real. Na figura A.1 apresentamos um gráfico do polinómio  $p$  para confirmar as nossas conclusões.*

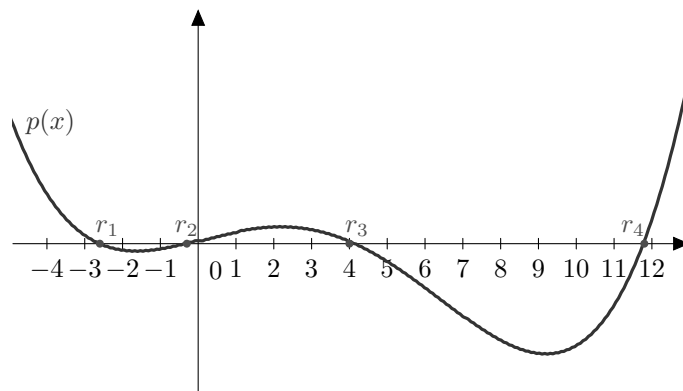


Figura A.1: Representação gráfica do polinómio  $p(x)$

Estes métodos são fáceis de trabalhar e dão, relativamente bons resultados. Em [19, secção 5.9] ou [4, secção 5.1–5.4] podemos ver estes e outros métodos que poderão ser aí consultados.

# Apêndice B

## Rotinas MATLAB<sup>®</sup>

Ao longo deste trabalho foram apresentados exemplos, nomeadamente nos Capítulos 5 e 6, para os quais recorremos a algumas rotinas em MATLAB<sup>®</sup>. Estas rotinas não são muito elaboradas pois não haveria espaço para uma discussão sobre os resultados computacionais obtidos. Apresentámos apenas alguns exemplos de aplicação dos métodos estudados.

### B.1 Métodos Numéricos de Aproximação de Zeros de Polinómios

Para o Exemplo (5.1), em que aplicámos o Método de Newton-Horner ao polinómio  $p(x) = x^4 - 13x^3 + 3x^2 + 124x + 36$  criámos a rotina, que indicamos a seguir, onde as variáveis de entrada são

- $P$  – vetor com os coeficientes do polinómio,
- $x0$  – aproximação inicial,
- `num_iter` – número de iterações desejadas,

e a variável de saída é

- `zaprox` – aproximação da raiz do polinómio  $P$ , após `num_iter` iterações.

```
function [zaprox]=MetodoHorner2(P,x0,num_iter)
n=length(P);
for i=1:num_iter
    B=[1 -x0];
    [Qi,Ri]=deconv(P,B);
    [Ti,Rimais1]=deconv(Qi,B);
    x0=x0-Ri(n)/Rimais1(n-1);
end;
zaprox=x0;
```

Para a aplicação do Método de Müller ao mesmo polinómio, Exemplo (5.2), utilizámos uma rotina em que as variáveis de entrada são

- $p$  – vetor com os coeficientes do polinómio  $p(x)$ , em estudo,
- $x_0, x_1, x_2$  – aproximações iniciais,
- `num_iter` – número de iterações desejadas,

e a variável de saída é

- $xf$  – aproximação final.

```
function xf=metodo_Muller(p,x0,x1,x2,num_iter)
X=[x0 x1 x2];
format long
for i=1:num_iter
    pX=polyval(p,X)
    v=[X(1)-X(3) X(2)-X(3) 0];
    V=vander(v)
    coef=V\pX'
    xf=X(3)-2*coef(3)/(coef(2)+sign(coef(2))*sqrt(coef(2)^2-4*coef(1)*coef(3)))
    X=[X(2) X(3) xf];
end
```

Finalmente, para a aplicação do Método de Bairstow, no Exemplo (5.3), criámos e utilizámos a rotina `metodo_bairstow`, cujas variáveis de entrada são

- $P$  – vetor com os coeficientes do polinómio  $p(x)$ ,
- `alpha0` – aproximação de  $\alpha$ ,
- `beta0` – aproximação de  $\beta$ ,

e cujas variáveis de saída são

- `alpha` – nova aproximação de  $\alpha$ ,
- `beta` – nova aproximação de  $\beta$ ,
- `resto` – resto da divisão do polinómio  $p(x)$  por  $(x^2 + \alpha x + \beta)$ .

```
function [alpha,beta,resto]=metodo_bairstow(P,alpha0,beta0)
format long
%DETERMINAÇÃO DOS COEFICIENTES bi
n=length(P);
```



```

B=zeros(size(P));
B(1)=P(1);
B(2)=P(2)-alpha0*B(1);
for i=3:n-1
    B(i)=P(i)-alpha0*B(i-1)-beta0*B(i-2);
end
B(n)=P(n)-beta0*B(n-2);

%DETERMINAÇÃO DAS DERIVADAS DE b1 E b0 EM ORDEM A alpha0 E beta0
Dalp_aux=zeros(size(B));
Dbeta_aux=zeros(size(B));

Dalp_aux(2)=-B(1);
for i=3:n-1
    Dalp_aux(i)=-B(i-1)-alpha0*Dalp_aux(i-1)-beta0*Dalp_aux(i-2);
    Dbeta_aux(i)=-alpha0*Dbeta_aux(i-1)-B(i-2)-beta0*Dbeta_aux(i-2);
end
Dalp_aux(n)=-beta0*Dalp_aux(n-2);
Dbeta_aux(n)=-B(n-2)-beta0*Dbeta_aux(n-2);

```

## B.2 Métodos Matriciais Iterativos de Aproximação de Valores Próprios

Para o Método das Potências criamos duas rotinas diferentes. Uma seguindo a vertente dada pelo autor Alfion Quarteroni em [20, pp. 192–194], `MPDireta_QUART`, e outra seguindo a abordagem do autor Demidovich em [4, pp. 430–436], `metodo_da_potencia_DEMI`. Em ambas as rotinas as variáveis de entrada são

- $A$  – matriz,
- $y_0$  – vetor inicial;
- `num_iter` – número de iterações desejadas,

e a variável de saída é

- `lambda1` – valor próprio de maior valor absoluto, aproximado.

Assim, nos Exemplos (6.1) e (6.2) as rotinas usadas foram as seguintes

```

function [lambda1,x1]=MPDireta_QUART(A,y0,num_iter)
    format long
    n=length(A);
    Y=zeros(n,num_iter);

    %iterações
    for i=1:1:num_iter

```

```

        yi=A^(i)*y0'/norm(A^(i)*y0');
        Y(:,i)=yi;
    end;
    lambda1=Y(:,num_iter)'*A*Y(:,num_iter);

```

e

```

function [lamda1,x1]=metodo_da_potencia_DEMI(A,y0,num_iter)
    n=length(A);
    Y=zeros(n,num_iter);

    %iterações
    for i=1:1:num_iter
        yi=(A^(i))*y0;
        Y(:,i)=yi;
    end;

    %cálculo da aproximação do valor próprio
    lamda_aux=zeros(1,n);
    for j=1:1:n
        lambda1j=Y(j,num_iter)/Y(j,num_iter-1);
        lambda_aux(j)=lambda1j;
    end;

    lamda1=sum(lambda_aux)/n;

```

No Exemplo (6.4), onde aplicámos o Método das Potências Inversas, utilizámos uma rotina cujas variáveis de entrada, para além das já citadas para o Método das Potências Diretas, inclui

- $\mu$  – escalar, aproximação de um qualquer valor próprio,

e a variável de saída é

- $\lambda_{\text{dban}}$  – valor próprio, aproximado a  $\mu$ .

```

function lambda_dban=MPInversa_QUART2(A,y0,mu,num_iter)
    n=length(A);
    Y=zeros(n,num_iter);

    %iterações
    for i=1:1:num_iter
        yi=(inv(A-mu*eye(n)))^i*y0'/norm((inv(A-mu*eye(n)))^i*y0');
        Y(:,i)=yi;
    end;
    lambda=Y(:,num_iter)'*(inv(A-mu*eye(n)))*Y(:,num_iter);
    lambda_dban=mu+1/lambda;

```

Para o Método de Jacobi criámos uma rotina, que aplicámos no Exemplo(6.12), em que a variável de entrada é

- $A$  – matriz simétrica,

e a variável de saída é

- $A1$  – nova matriz resultante de uma iteração do método.

```
function [A1]=metodo_Jacobi(A)
format long
%Construção de A_aux – matriz auxiliar para encontrar
%o valor máximo fora da diagonal
n=size(A);
A_aux=A;

for i=1:n
    A_aux(i:n,i)=0;
end

%Determinação de p,q – índices da posição do valor máximo, em módulo
%Índice da linha
[Y1 p]=max(max(abs(A_aux')));
%Índice da coluna
[Y2 q]=max(max(abs(A_aux)));

%Cálculos auxiliares
a=(A(p,p)-A(q,q))/(2*A(p,q));
t=sign(a)/(abs(a)+sqrt(a^2+1));
c=1/sqrt(t^2+1);
s=t*c;

%Construção da matriz de Givens
G=eye(n);
G(p,p)=c;
G(q,q)=c;
G(p,q)=-s;
G(q,p)=s;

%iteração
A1=G'*A*G;
```



# Bibliografia

- [1] Beezer, Robert A. *A First Course in Linear Algebra*. Department of Mathematics and Computer Science, University of Puget Sound, Washington, USA , 2004.
- [2] Bhatia, Rajendra. *Matrix Analysis*. Springer-Verlag New York Inc., USA , 1997.
- [3] Burden, Richard L. & Faires, J. Douglas. *Numerical Analysis*. 5th ed. Boston, MA: PWS Publishing Company. London: ITP International Thomson Publishing, 1993.
- [4] Demidovic, B. P. & Maron, I. A. *Computational Mathematics*. Translated from the Russian by George Yankovsky. Moscow: Mir Publishers, 1976.
- [5] Fernandes, Edite Manuela da G. P. *Computação Numérica*. 2.nd ed. Universidade do Minho, Braga, 1997.
- [6] Fernández, P. El Secreto de Google® y el Álgebra lineal. *Boletín de la Sociedad Española de Matemática Aplicada*, 30, 2004.
- [7] Gantmacher, F. R. *Applications of the Theory of Matrices*. Interscience Publishers, Inc., New York, 1959.
- [8] Golub, Gene H. & Van Loan, Charles F. *Matrix Computations*, 3.rd ed. The Johns Hopkins University Press, Maryland, 1996.
- [9] Heij, C.; Ran, A. & Schagen, F. *Introduction to Mathematical Systems Theory. Linear Systems, Identification and Control*. Birkhäuser, 2007.
- [10] Horn, Roger A. & Johnson, Charles R. *Matrix analysis*. Cambridge University Press, 1985.
- [11] Householder, A. S. *The Theory of Matrices in Numerical Analysis*. A Blaisdell Book in the Pure and Applied Sciences. New York-Toronto-London: Blaisdell Publishing Company, a division of Ginn and Company, 1964.
- [12] Katz, Victor J. *História da Matemática*. Fundação Calouste Gulbenkian, Serviço de Educação e Bolsas., 2010.
- [13] Kiusalaas, Jaan. *Numerical Methods in Engineering with MATLAB®*. Cambridge University Press, 2005.

- [14] Kurosh, A. *Higher Algebra. Translated from the Russian by George Yankovsky. 4.th printing.* Moscow: Mir Publishers., 1984.
- [15] Lancaster, Peter & Tismenetsky, Miron. *The theory of matrices. 2nd ed., with applications.* Computer Science and Applied Mathematics. Orlando etc.: Academic Press (Harcourt Brace Jovanovich, Publishers), 1985.
- [16] Luzardo, Deivi & P., Alirio J. Peña. Historia del Álgebra Lineal hasta los Albores del Siglo XX. *Divulgaciones Matemáticas*, 14(2), 2006.
- [17] School of Mathematics and Scotland Statistics, University of St Andrews. Mathematicians born from 1700 to 1749. [http://www-history.mcs.st-and.ac.uk/Indexes/1700\\_1749.html](http://www-history.mcs.st-and.ac.uk/Indexes/1700_1749.html), 2012. [visitado em 20-Outubro-2012].
- [18] Pedroche, F. Métodos de cálculo del vector PageRank. *Boletín de la Sociedad Española de Matemática Aplicada*, 39, 2007.
- [19] Pina, Heitor. *Métodos Numéricos.* Escolar Editora: Lisboa, 2010.
- [20] Quarteroni, Alfio; Sacco, Riccardo & Saleri, Fausto. *Numerical mathematics. 2nd ed.* Texts in Applied Mathematics 37, Berlin: Springer, 2007.
- [21] Rao, Singiresu S. *Applied Numerical Methods for Engineers and Scientists.* Prentice Hall, 2002.
- [22] Serre, Denis. *Matrices. Theory and applications. (Les matrices: théorie et pratique.).* Masson Sciences, Mathématiques, Paris: Dunodi, 2001.
- [23] Singh, Gulshan & Shah, Mohammad. On the location of zeros of polynomials. *American Journal of Computational Mathematics*, 1(1), 2011.
- [24] Varadarajan, V. S. *Algebra in Ancient and Modern Times. – Mathematical World Vol.12.* American Mathematical Society - Hindustan Book Agency, 1998.
- [25] Wang, Xiaoshen. A Simple Proof of Decartes's Rule of Signs. *The American Mathematical Monthly*, 111(6), 2004.
- [26] Watkins, David S. The QR Algorithm Revisited. *SIAM Review*, 50(1), 2008.
- [27] Wayland, Harold. Expansion of Determinantal Equations Into Polynomial Form. *Quarterly of Applied Mathematics*, II(4), 1945.
- [28] Wikipedia. Eigenvalues and eigenvectors. [http://en.wikipedia.org/wiki/Eigenvalues\\_and\\_eigenvectors](http://en.wikipedia.org/wiki/Eigenvalues_and_eigenvectors), 2012. [visitado em 20-Outubro-2012].
- [29] Wilkinson, J. H. *The Algebraic Eigenvalue Problem.* Monographs on Numerical Analysis. Oxford: At the Clarendon Press, 1965.